Generalizing DP-SGD with Shuffling and Batch Clipping

Marten van Dijk[1,2,3], **Phuong Ha Nguyen**[4], **Toan N. Nguyen**[5†],
Lam M. Nguyen[6],
[1] CWI Amsterdam, The Netherlands
[2] Department of Computer Science, Vrije Universiteit Amsterdam, The Netherlands
[3] Department of Electrical and Computer Engineering, University of Connecticut, CT, USA
[4] eBay, CA, USA
[5] Department of Computer Science and Engineering, University of Connecticut, CT, USA
[6] IBM Research, Thomas J. Watson Research Center, Yorktown Heights, NY, USA

marten.van.dijk@cwi.nl, toan.nguyen@uconn.edu,
LamNguyen.MLTD@ibm.com, phuongha.ntu@gmail.com

Abstract

Classical differential private DP-SGD implements individual clipping with random subsampling, which forces a mini-batch SGD approach. We provide a general differential private algorithmic framework that goes beyond DP-SGD and allows any possible first order optimizers (e.g., classical SGD and momentum based SGD approaches) in combination with batch clipping, which clips an aggregate of computed gradients rather than summing clipped gradients (as is done in individual clipping). The framework also admits sampling techniques beyond random subsampling such as shuffling. Our DP analysis follows the f-DP approach and introduces a new proof technique based on a slightly *stronger* adversarial model which allows us to derive simple closed form expressions and to also analyse group privacy. In particular, for E epochs work and groups of size g, we show a $\sqrt{gE}$ DP dependency for batch clipping with shuffling.

1 Introduction

In order to defend against privacy leakage of local proprietary data during collaborative training of a global model as in federated learning [12], [1] introduced DP-SGD as it adapts distributed Stochastic Gradient Descent (SGD)[13, 14] with Differential Privacy (DP)[6, 4, 7, 5].

This approach allows each client to perform local SGD computations for the data samples ξ that belong to the client's local training data set d. Each client is doing SGD computations for a batch of local data. These recursions together represent a local round and at the end of the local round a local model update (in the form of an aggregate of computed gradients during the round) is transmitted to the server. The server in turn adds the received local update to its global model – and once the server receives new updates from (a significant portion of) all clients, the global model is broadcast to each of the clients. When considering privacy, we are concerned about how much information these local updates reveal about the used local data sets.

Depicted in Algorithm 1 (a detailed description and explanation is given in Section 2), we introduce a general algorithmic framework of which DP-SGD is one example. The main generalization beyond DP-SGD is that the most inner loop can execute any optimization algorithm $\mathcal{A}$ in order to compute a local round update U. DP-SGD is realized for $s = 1$ with

[†] Supported by NSF grant CNS-1413996 "MACS: A Modular Approach to Cloud Security."

$\mathcal{A}$ computing a single gradient:

$$U = \sum_{i \in S_b} [\nabla f(w, \xi_i)]_C, \tag{1}$$

where $[x]_C = x/\max\{1, \|x\|/C\}$ is the clipping operation and S_b is a batch sampled from local data set d of size m.

We call (1) the *individual clipping* approach and notice that it implements mini-batch SGD. This is necessary because the DP analysis requires each clipped output to be independent from other clipped outputs. This implies that computation of one clipped output should not involve updating a local model w which is used in a computation of another next clipped output. This means that each of the clipped gradients in the sum are evaluated in the same w (the most recently received global model from the server). We notice that the clipping operation introduces 'clipping' noise and together with the added Gaussian noise for differential privacy this results in convergence to a final global model with smaller (test) accuracy (than what otherwise, without DP, can be achieved).

Our algorithmic framework allows a much wider choice. In particular, $m = 1$ allows for example

$$U = [\sum_{j=1}^{s} \nabla f(w_j, \xi_{i_j})]_C, \tag{2}$$

where $S_b = \{i_1, \ldots, i_s\}$ is a batch sampled from the local data set d of size s and where algorithm $\mathcal{A}$ implements classical SGD according to the SGD recursion

$$w_{j+1} = w_j - \eta \nabla f(w_j, \xi_{i_j}),$$

where $w_1 = w$ is initialized by the most recently received global model from the server (possibly updated with updates send to the central server, see lines 8 and 24 in Algorithm 1) and where $\eta = \eta_{(e-1)\frac{N}{ms}+b}$ is the round step size (for round b in epoch e).

We call (2) an example of *batch clipping*. Batch clipping in its most general form is

$$U = [\mathcal{A}(w, \{\xi_i\}_{i \in S_b})]_C. \tag{3}$$

It allows us to implement classical SGD and go beyond mini-batch SGD such as Adam [8], AdaGrad [3], SGD-Momentum [17], RMSProp [20]. The framework is general in that $\mathcal{A}$ can implement any optimization algorithm including classical (as discussed above) or momentum based SGD type algorithms.

We have two main contributions:

General Algorithmic Framework: Algorithm 1 with detailed discussion in Section 2 defines our general framework. In the inner loop it allows execution of any optimization algorithm $\mathcal{A}$ of our choice. This covers classical SGD and momentum based SGD, and can possibly be extended with batch normalization (over S_b). Our framework is compatible with asynchronous communication between clients and server. We also notice that the framework allows diminishing step sizes (learning rate), and allows adapting the clipping constant from round to round (this is compatible with DP analysis in general).

DP Proof Technique based on a Sampling Induced Distribution: We introduce a *slightly* stronger adversary, denoted by $\mathcal{A}_1$ in Table 1, compared to the model, denoted by $\mathcal{A}_0$ in Table 1, used in DP proofs in current literature. The adversary knows how Sample_m operates and can derive a joint probability distribution $\mathbb{P}$ of the number of differentiating data samples for each round within the sequence of rounds that define the series of epochs during which updates are computed. DP analysis based on $\mathcal{A}_0$ considers each round separately and analysis of a single round assumes a strong adversary who knows the instance drawn from the projection of $\mathbb{P}$ on that round (analysis of the whole sequence of rounds follows from composition of the derived DP guarantees of single rounds). Our DP analysis based on $\mathcal{A}_1$ deals with the whole sequence of rounds at once and the resulting analysis assumes a slightly stronger adversary who knows the instance drawn from $\mathbb{P}$. Appendix B discusses the adversarial models in more detail.

We introduce a probability distribution $q_E(c)$ induced by the sampling procedure $\mathsf{Sample}_{s,m}$. Together with the knowledge of an instance of $\mathbb{P}$ given to our slight stronger adversary in our DP analysis, we prove an f-DP guarantee [2] where f is related to a mix of Gaussian trade-off functions $G_{c/\sigma}$ according to distribution $q_E(c)$.

Table 1 summarizes our results which we interpret in Section 4. Our DP analysis shows that a much wider class of SGD based algorithms have provable DP guarantees. We remove the constraints made in the original DP-SGD paper; we can go beyond mini-batch SGD and we may replace random subsampling by shuffling. We see that in f-DP terminology various configurations in the general algorithmic framework provide a $\approx G_{\sqrt{gE}/\sigma}$-DP guarantee with a $\sqrt{gE}$ dependency, where we consider group privacy for groups of size g, E is the total number of epochs (measured in data set size N) worth of gradient computations, and σ characterizes the added Gaussian DP noise (after normalization

Algorithm 1 Generalized Framework for DP-SGD

```
procedure DP-SGD-GENERAL
    $N =$ size training data set $d = \{\xi_i\}_{i=1}^N$
    $E =$ total number of epochs
    $T =$ total number of rounds
    diminishing round step size sequence $\{\eta_i\}_{i=1}^T$

    initialize $w$ as the default initial model
    Interrupt Service Routine (ISR): Whenever a new global model $\hat{w}$ is received, computation is interrupted
  and an ISR is called that replaces $w \leftarrow \hat{w}$ after which computation is resumed

    for $e \in \{1, \ldots, E\}$ do
        Let $\pi^e$ be a random permutation
        re-index data samples: $\{\xi_i \leftarrow \xi_{\pi^e(i)}\}_{i=1}^N$
        $\{S_{b,h}\}_{b=1,h=1}^{N/(ms),m} \leftarrow \texttt{Sample}_{s,m}$ with
            $S_{b,h} \subseteq \{1, \ldots, N\}$,
            $|S_{b,h}| = s$, $|S_b| = sm$ with $S_b = \bigcup_{h=1}^m S_{b,h}$
        for $b \in \{1, \ldots, \frac{N}{ms}\}$ do
            Start of round $(e-1)\frac{N}{ms} + b$:
            for $h \in \{1, \ldots m\}$ do
                $a_h \leftarrow \mathcal{A}(w, \{\xi_i\}_{i \in S_{b,h}})$
            end for
            $U = \sum_{h=1}^m [a_h]_C$
            $\bar{U} \leftarrow U + \mathcal{N}(0, (2C\sigma)^2\mathbf{I})$
            Transmit $\bar{U}/m$ to central server
            Locally update $w \leftarrow w - \eta_{(e-1)\frac{N}{ms}+b} \cdot \bar{U}/m$
        end for
    end for
end procedure
```

with the clipping constant). This square root dependency, in particular for group privacy (here shown for the first time), must as a consequence also hold for the weaker $\mathcal{A}_0$ adversary.

Outline: Section 2 defines our general algorithmic framework. The minimal necessary background on f-DP is in Section 3 (with a full description in A). Section 4 states the main definitions and main theorems (proved in Appendices C and D with a discussion on the used adversarial model in Appendix B) together with a detailed discussion of the application of the main theorems in Table 1 (with corresponding proofs/applications for the tabled cases in Appendices F and G). Section 5 shows simulations for mini-batch SGD with individual clipping, batch clipping, and a mixed clipping that merges both approaches. The accuracy of each are comparable. The advantage of using batch clipping over individual and mixed clipping is improved memory utilization. This shows that the full extent/flexibility of our framework is worth studying in future work.

2 Algorithmic Framework for Differential Private SGD

We provide a general algorithmic framework for differential private SGD in Algorithm 1:

Asynchronous SGD: We allow asynchronous communication between clients and the central aggregating server. Each client maintains its own local model. Each client implements an Interrupt Service Routine (ISR) which replaces its local model w with any newly received global model $\hat{w}$ from the server. In practice asynchronous communication may lead to out of order arrival or dropped global models broadcast by the central server. A limited amount of asynchronous behavior leads to provably optimal convergence rates for certain learning tasks [15].

Sampling Strategies: The whole training process spans E epochs. At the start of each epoch $\texttt{Sample}_{s,m}$ generates sets $\{S_{b,h} \subseteq \{1, \ldots, N\}\}_{b=1,h=1}^{N/(ms),m}$ where each set $S_{b,h}$ has size s; we define $\{S_b = \bigcup_{h=1}^m S_{b,h}\}_{b=1}^{N/(ms)}$ where each set S_b has size ms. Here, b indexes the round within an epoch during which the client uses its training data to compute a local update that is transmitted to the server. After multiplying with the appropriate learning rates (step sizes η_i), the

ind. clipping (1)	batch clipping (2), (3)	gen. clipping (4), (6)	subsampling	shuffling	group privacy	h-DP guarantee	adversarial model
x			x		$g=1$	$h = C_{m/N}(G_{1/\sigma})^{\otimes (N/m)\cdot E}$ from [2], the *original DP-SGD setting* [1]	$\mathcal{A}_0$
x			x		$g \geq 1$	h is in the range $\approx \left[G_{\sqrt{(e+\gamma)\min\{m,g\}gE}/\sigma}, (\alpha \rightarrow 1-\alpha)\right]$, approximation is for small $e^{-\gamma g E}$	$\mathcal{A}_1$
	x		x		$g \geq 1$	h is in the range $\approx \left[G_{\sqrt{(1+1/\sqrt{2gE})gE}/\sigma}, G_{\sqrt{(1-1/\sqrt{2gE})gE}/\sigma}\right]$, approximation is for small e^{-gE} and small s/N (see Thm G.1 for a tighter lower bound corresponding to Def 4.2)	$\mathcal{A}_1$
		x	x		$g=1$	h is in the range $\approx \left[G_{\sqrt{(1+1/\sqrt{2E})E}/\sigma}, G_{\sqrt{(1-1/\sqrt{2E})E}/\sigma}\right]$, approximation is for small e^{-E}	$\mathcal{A}_1$
		x		x	$g=1$	$h = G_{\sqrt{E}/\sigma}$	$\mathcal{A}_1$
		x		x	$g \geq 1$	h is in the range $\approx [G_{\sqrt{gE}/\sigma}, (\alpha \rightarrow 1-\alpha)]$, approximation is for constant E and small $g^2 ms/(N-g)$	$\mathcal{A}_1$
	x			x	$g \geq 1$	$h \in \left[G_{\sqrt{gE}/\sigma}, (\alpha \rightarrow 1-\alpha)\right]$	$\mathcal{A}_1$
	x			x	$g \leq s$	$h \approx G_{\sqrt{gE}/\sigma}$, approximation is for constant E and small $g^2/(N/s-g-g^2)$	$\mathcal{A}_1$

Table 1: Trade-off functions h for the mechanism $\mathcal{M}$ defined by Algorithm 1 in the strong ($\mathcal{A}_0$) or slightly stronger ($\mathcal{A}_1$) adversarial model for all pairs of data sets d and d' with $\max\{|d \setminus d'|, |d' \setminus d|\} \leq g$; h cannot be improved beyond the reported ranges closer towards function $\alpha \rightarrow 1-\alpha$ (which represents random guessing of the hypothesis d or d', hence, no privacy leakage). An approximation for a small quantity means that if the quantity tends to 0, then the approximation becomes tight.

server aggregates received local updates from different clients into the global model maintained by the server. Within round b, h indexes independent computations by an algorithm $\mathcal{A}$ (discussed later). Each computation starts from the client's local model. The h-th computation is based on training data samples from data set d that corresponds to $S_{b,h}$.

In this paper we analyze and compare two distinct sampling strategies for $\texttt{Sample}_{s,m}$:

- **Subsampling (SS):** Sets $S_b \subseteq \{1, \ldots, N\}$ with $|S_b| = ms$ are randomly sampled from $\{1, \ldots, N\}$.
- **Shuffling (SH):** The whole index set $\{1, \ldots, N\}$ is shuffled into a random order (according to a random chosen permutation). This is used to partition data set d into sets $S_b \subseteq \{1, \ldots, N\}$ with $|S_b| = ms$ in a random way. Notice that $d = \bigcup_{b=1}^{N/(ms)} S_b$.

The difference between the two strategies is that the S_b are disjoint for shuffling while they may have intersections for subsampling. Once the S_b are sampled, $\texttt{Sample}_{s,m}$ selects a random chosen partition in order to split S_b into subsets $S_{b,h}$, $1 \leq h \leq m$, with $|S_{b,h}| = s$.

Update Algorithm: A round computes some partial update $a_h \leftarrow \mathcal{A}(w, \{\xi_i\}_{i \in S_{b,h}})$, which only depends on w and the set of training data samples $S_{b,h}$. Here, w is either equal to the most recent received global model or has been locally updated using some step size in combination with a local noised aggregated update $\bar{U}$ that was also transmitted to the central server. This means that any observer of communication with the central server is also able to compute the used w based on previously observed updates $\bar{U}$.

Algorithm $\mathcal{A}$ considers in sequence each of the s training samples in $\{\xi_{i_1}, \ldots, \xi_{i_s}\}$, where $S_{b,h} = \{i_1, \ldots, i_s\}$ is the random index set produced by $\texttt{Sample}_{s,m}$. We notice that our algorithmic framework and DP analysis allow any other optimization algorithm such as momentum based SGD. It may include batch normalization where normalization is done over the batch $\{\xi_{i_1}, \ldots, \xi_{i_s}\}$, and it may also include instance or layer normalization. We may even predefine a sequence of algorithms $\{\mathcal{A}_b\}_{b=1}^{N/(ms)}$ per epoch and use $\mathcal{A}_b$ in round b.

Clipping: To each computed a_h we apply clipping $x \rightarrow [x]_C = x/\max\{1, \|x\|/C\}$. We aggregate the clipped $[a_h]_C$ in a sum U. We add Gaussian noise before it is transmitted to the central server. Each round reveals one noisy update U and we want to bound the aggregated differential privacy leakage. The general formula for U is

$$U = \sum_{h=1}^{m} [a_h]_C = \sum_{h=1}^{m} [\mathcal{A}(w, \{\xi_i\}_{i \in S_{b,h}})]_C. \tag{4}$$

Gaussian Noise: After U is computed according to (4), Gaussian noise $\mathcal{N}(0, (2C\sigma)^2)$ is added to each entry of vector U. The resulting noisy update $\bar{U}$ after averaging by m is sent to the central server.

Some Remarks: We notice that the framework allows a diminishing step size sequence (learning rate). The DP analysis shows that we may adapt the clipping constant at the start of each round. Rather than computing U as the sum $\sum_{h=1}^{m} [a_h]_C$, we may compute $\sum_{h=1}^{m} \mathcal{B}([a_h]_{C'})$ for some post-processing function/procedure $\mathcal{B}$ (here, we need to take care that $\|\mathcal{B}(x)\| \leq C$ for $\|x\| \leq C'$). By revealing a differential private noisy mean and noisy variance of the training data set (due to differential private data normalization pre-processing), algorithm $\mathcal{A}$ can implement data normalization; in the f-DP framework, privacy leakage is now characterized as a trade-off function of the differential private data normalization pre-processing composed with the trade-off function corresponding to our DP analysis for Algorithm 1. Without loss of differential privacy, we may under certain circumstances (by changing sampling to batches S_b with non-fixed probabilistic sizes) use Gaussian noise $\mathcal{N}(0, (C\sigma)^2\mathbf{I})$, a factor 2 less; a discussion is in Appendix E.

3 Background f-DP

Dong et al. [2] introduced the state-of-the-art DP formulation based on hypothesis testing. From the attacker's perspective, it is natural to formulate the problem of distinguishing two neighboring data sets d and d' based on the output of a DP mechanism $\mathcal{M}$ as a hypothesis testing problem:

$$\begin{array}{ll} & H_0 : \text{ the underlying data set is } d \\ \text{versus} & H_1 : \text{ the underlying data set is } d'. \end{array}$$

Here, neighboring means that either $|d \setminus d'| = 1$ or $|d' \setminus d| = 1$. More precisely, in the context of mechanism $\mathcal{M}$, $\mathcal{M}(d)$ and $\mathcal{M}(d')$ take as input representations r and r' of data sets d and d' which are 'neighbors.' The representations are mappings from a set of indices to data samples with the property that if $r(i) \in d \cap d'$ or $r'(i) \in d \cap d'$, then $r(i) = r'(i)$. This means that the mapping from indices to data samples in $d \cap d'$ is the same for the representation of d and the representation of d'. In other words the mapping from indices to data samples for d and d' only differ for indices corresponding to the differentiating data samples in $(d \setminus d') \cup (d' \setminus d)$. In this sense the two mappings (data set representations) are neighbors. In our main theorem we will consider the general case $g = \max\{|d \setminus d'|, |d' \setminus d|\}$ in order to analyse 'group privacy.'

We define the Type I and Type II errors by

$$\alpha_\phi = \mathbb{E}_{o \sim \mathcal{M}(d)}[\phi(o)] \text{ and } \beta_\phi = 1 - \mathbb{E}_{o \sim \mathcal{M}(d')}[\phi(o)],$$

where ϕ in $[0, 1]$ denotes the rejection rule which takes the output of the DP mechanism as input. We flip a coin and reject the null hypothesis with probability ϕ. The optimal trade-off between Type I and Type II errors is given by the trade-off function

$$T(\mathcal{M}(d), \mathcal{M}(d'))(\alpha) = \inf_\phi \{\beta_\phi \ : \ \alpha_\phi \leq \alpha\},$$

for $\alpha \in [0, 1]$, where the infimum is taken over all measurable rejection rules ϕ. If the two hypothesis are fully indistinguishable, then this leads to the trade-off function $1 - \alpha$. We say a function $f \in [0, 1] \rightarrow [0, 1]$ is a trade-off function if and only if it is convex, continuous, non-increasing, at least 0, and $f(x) \leq 1 - x$ for $x \in [0, 1]$.

We define a mechanism $\mathcal{M}$ to be f-DP if f is a trade-off function and

$$T(\mathcal{M}(d), \mathcal{M}(d')) \geq f$$

for all neighboring d and d'. The f-DP framework supersedes all existing other frameworks in that a trade-off function contains all the information needed to derive known DP metrics such as (δ, ϵ)-DP and divergence based DPs.

[2] defines Gaussian DP as a special case of f-DP where f is a trade-off function

$$G_\mu(\alpha) = T(\mathcal{N}(0, 1), \mathcal{N}(\mu, 1))(\alpha) = \Phi(\Phi^{-1}(1 - \alpha) - \mu)$$

with Φ the standard normal cumulative distribution of $\mathcal{N}(0,1)$.

The tensor product $f \otimes h$ for trade-off functions $f = T(P,Q)$ and $h = T(P',Q')$ is well-defined by

$$f \otimes h = T(P \times P', Q \times Q').$$

Let $y_i \leftarrow \mathcal{M}_i(\mathtt{aux}, d)$ with $\mathtt{aux} = (y_1, \ldots, y_{i-1})$. Theorem 3.2 in [2] shows that if $\mathcal{M}_i(\mathtt{aux}, .)$ is f_i-DP for all $\mathtt{aux}$, then the composed mechanism $\mathcal{M}$, which applies $\mathcal{M}_i$ in sequential order from $i = 1$ to $i = T$, is $(f_1 \otimes \ldots \otimes f_T)$-DP. The tensor product is commutative. As a special case Corollary 3.3 in [2] states that composition of multiple Gaussian operators G_{μ_i} results in G_μ where $\mu = \sqrt{\sum_i \mu_i^2}$.

Suppose that d and d' do not differ in just one sample, but differ in g samples. [2] shows that if a mechanism is G_μ-DP, then it is $G_{g\mu}$-DP for groups of size g. This shows a linear dependency in g.

A full description of f-DP (including the subsampling operator $C_{m/N}$) with intuitive explanation is in Appendix A.

4 DP Analysis

Our main theorems, stated below, are proved in Appendices C and D. We consider two data sets d and d' that differ in g samples in order to also analyze group privacy.

The main idea is to bound the sensitivity[*] of $\sum_{h=1}^m [a_h]_C$ in Algorithm 1 by $2kC$ where k is equal to the number of a_h whose computation depends on differentiating samples (samples outside $d \cap d'$). The number of rounds b within an epoch that have k values a_h that depend on differentiating samples is denoted by c_k. An epoch instance is into some extent characterized by the vector $(c_1, c_2, \ldots, c_g)$ where c_k indicates the number of rounds that have sensitivity equal to $2kC$.

If a round has a sensitivity $2kC$, then the trade-off function of its corresponding mechanism, after adding Gaussian noise $\mathcal{N}(0, (2C\sigma)^2\mathbf{I})$, is given by the Gaussian trade-off function $G_{k/\sigma}$. Therefore, an epoch instance $(c_1, c_2, \ldots, c_g)$ has a Gaussian trade-off function which is the composition of the round trade-off functions $G_{k/\sigma}$: We have

$$G_{1/\sigma}^{\otimes c_1} \otimes G_{2/\sigma}^{\otimes c_2} \otimes \ldots \otimes G_{g/\sigma}^{\otimes c_g} = G_{\sqrt{\sum_{k=1}^g c_k k^2}/\sigma}.$$

This can also be composed over multiple epochs. This leads to defining a probability distribution $q_E(c)$ which is the probability that E epoch instances together realize the value $c = \sqrt{\sum_{k=1}^g c_k k^2}$. With probability $q_E(c)$, mechanism $\mathcal{M}$ executes a 'sub-mechanism' which has trade-off function $G_{c/\sigma}$. The trade-off function of the overall mechanism $\mathcal{M}$ is therefore related, but not exactly equal, to the expectation $\sum_c q_E(c) \cdot G_{c/\sigma}$: See Lemma C.1 in Appendix C, we have $T(\mathcal{M}(d), \mathcal{M}(d')) \geq f$, where $f(\alpha)$ equals the trade-off function

$$\inf_{\{\alpha_c\}} \left\{ \sum_c q_E(c) G_{c/\sigma}(\alpha_c) \;\middle|\; \sum_i q_E(c)\alpha_c = \alpha \right\}. \tag{5}$$

Random variable c_k counts the number of rounds within the E epochs that have sensitivity $2kC$. The vector of random variables $(c_1, \ldots, c_g)$ defines the random variable $c = \sqrt{\sum_{k=1}^g c_k k^2}$. By analyzing its probability distribution $q_E(c)$ we are able to derive upper and lower bounds for f defined in (5). The theorem provides a solution $f(\alpha)$ for the infinum and provides lower and upper bound functions. This leads to $T(\mathcal{M}(d), \mathcal{M}(d') \geq f$ with upper and lower bounds which, see Table 1, can be very close to one another.

We first define/formalize a couple of concepts before stating our main theorems (proofs are in Appendices C and D).

Definition 4.1. Let $\mathcal{M}$ be the mechanism corresponding to our general framework for E epochs which is based on $\mathsf{Sample}_{s,m}$ with N equal to the size of the data set that is sampled. We require that if $\mathtt{Sample}_{s,m}$ outputs $\{S_{b,h}^e\}_{b=1,h=1}^{N/(sm),m}$ for the e-th epoch, then for each b, $\{S_{b,h}^e\}_{h=1}^m$ partitions a set of size ms into m subsets $S_{b,h}^e$ of size s. Define

$$\mathcal{C} = \left\{ \sqrt{\sum_{k=1}^g c_k k^2} \;:\; \forall_k \; c_k \in \mathbb{N} \right\}.$$

[*]The sensitivity of a value is the Euclidean norm between its evaluation for d and d' respectively.

The sampling procedure $\mathtt{Sample}_{s,m}$ defines a probability distribution $q_E(c)$ over $\mathcal{C}$ as follows (notice that $q_E(c)$ implicitly depends on N):

$$q_E(c) =$$
$$\mathtt{Pr}_{\{\pi^e\}}\left[\begin{array}{c} c^2 = \sum_{k=1}^{g} c_k k^2 \\ \text{with } c_k = \#\{(b,e) : L_{b,e} = k\} \text{ and} \\ L_{b,e} = |\{h \, : \, \pi^e(S^e_{b,h}) \cap \{1,\ldots,g\} \neq \emptyset\}| \\ \text{conditioned on} \\ \{\{S^e_{b,h}\}_{b=1,h=1}^{N/(sm),m} \leftarrow \mathtt{Sample}_{s,m}\}_{e=1}^{E} \end{array}\right].$$

The next definitions define three trade-off functions based on $q_E(c)$ (which, as a consequence, also depend on N). The proof of the main theorems in Appendix D show that the functions are well-defined.

Definition 4.2. For distribution $q_E(c)$ over $c \in \mathcal{C}$ we define for $\alpha \in [0,1]$ function

$$f(\alpha) = \sum_{c \in \mathcal{C}} q_E(c) \cdot \Phi\left(\Lambda(\alpha) \cdot \frac{\sigma}{c} - \frac{c}{2\sigma}\right),$$

where function $\Lambda(\alpha)$, $\alpha \in [0,1]$, is implicitly defined by

$$1 - \alpha = \sum_{c \in \mathcal{C}} q_E(c) \cdot \Phi\left(\Lambda(\alpha) \cdot \frac{\sigma}{c} + \frac{c}{2\sigma}\right).$$

Definition 4.3. Let $q_E(c)$ be a distribution over $c \in \mathcal{C}$. Let $u_{c^{\mathtt{U}}}$ and $c^{\mathtt{U}}$ be such that

$$1 \geq u_{c^{\mathtt{U}}} \geq \sum_{c \in \mathcal{C} : c < c^{\mathtt{U}}} q_E(c).$$

Notice that $u_{c^{\mathtt{U}}}$ bounds the tail of $q_E(c)$ left from $c^{\mathtt{U}}$.

For $\alpha \in [0,1]$ we define functions

$$\begin{array}{rcl} \hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}(\alpha) & = & u_{c^{\mathtt{U}}} + (1 - u_{c^{\mathtt{U}}}) G_{c^{\mathtt{U}}/\sigma}(\alpha), \\ f^{\mathtt{U}}_{c^{\mathtt{U}}}(\alpha) & = & \min\{\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}, {\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}}^{-1}\}^{**}(\alpha) \\ & = & \left\{ \begin{array}{ll} u_{c^{\mathtt{U}}} + (1 - u_{c^{\mathtt{U}}}) G_{c^{\mathtt{U}}/\sigma}(\alpha), & \alpha \in [0, \beta_0], \\ \beta_0 + \beta_1 - \alpha, & \alpha \in [\beta_0, \beta_1], \\ G_{c^{\mathtt{U}}/\sigma}(\frac{\alpha - u_{c^{\mathtt{U}}}}{1 - u_{c^{\mathtt{U}}}}), & \alpha \in [\beta_1, 1], \end{array} \right. \end{array}$$

where

$$\begin{array}{rcl} \beta_0 & = & 1 - \Phi(\frac{c^{\mathtt{U}}}{2\sigma} - \frac{\sigma}{c^{\mathtt{U}}} \ln(1 - u_{c^{\mathtt{U}}})), \\ \beta_1 & = & u_{c^{\mathtt{U}}} + (1 - u_{c^{\mathtt{U}}}) \cdot (1 - \Phi(\frac{c^{\mathtt{U}}}{2\sigma} + \frac{\sigma}{c^{\mathtt{U}}} \ln(1 - u_{c^{\mathtt{U}}}))). \end{array}$$

Definition 4.4. Let $q_E(c)$ be a distribution over $c \in \mathcal{C}$. Let $l_{c^{\mathtt{L}}}$ and $c^{\mathtt{L}}$ be such that

$$1 \geq l_{c^{\mathtt{L}}} \geq \sum_{c \in \mathcal{C} : c > c^{\mathtt{L}}} q_E(c).$$

Notice that $l_{c^{\mathtt{L}}}$ bounds the tail of $q_E(c)$ right from $c^{\mathtt{L}}$.

For $\alpha \in [0,1]$ we define functions

$$\begin{array}{rcl} \hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}(\alpha) & = & G_{c^{\mathtt{L}}/\sigma}(\min\{1, \alpha + l_{c^{\mathtt{L}}}\}), \\ f^{\mathtt{L}}_{c^{\mathtt{L}}}(\alpha) & = & \max\{\hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}, {\hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}}^{-1}\}(\alpha) \\ & = & \left\{ \begin{array}{ll} G_{c^{\mathtt{L}}/\sigma}(\alpha) - l_{c^{\mathtt{L}}}, & \alpha \in [0, \beta], \\ G_{c^{\mathtt{L}}/\sigma}(\min\{1, \alpha + l_{c^{\mathtt{L}}}\}), & \alpha \in [\beta, 1], \end{array} \right. \end{array}$$

where $\beta \in [0,1]$ is implicitly defined as the (unique) solution of

$$G_{c^{\mathtt{L}}/\sigma}(\beta) - l_{c^{\mathtt{L}}} = \beta$$

(and we notice that ${\hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}}^{-1}(\beta) = \beta = \hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}(\beta)$).

Theorem 4.5. *If $\mathcal{M}$ is h-DP for all pairs of data sets d and d' with* $\max\{|d \setminus d'|, |d' \setminus d|\} \leq g$ *and* $N = |d \cap d'| + g$*, then*

$$h \leq f^{\mathtt{U}}_{c^{\mathtt{U}}} \leq \hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}$$

in adversarial model $\mathcal{A}_1$. We notice that $f^{\mathtt{U}}_{c^{\mathtt{U}}}$ is a symmetric trade-off function.

Theorem 4.6. *In adversarial model $\mathcal{A}_1$, $\mathcal{M}$ is f-DP for all pairs of data sets d and d' with* $\max\{|d \setminus d'|, |d' \setminus d|\} \leq g$ *and $N = |d \cap d'| + g$. Trade-off function f has lower bound*

$$f \geq f^{\mathtt{L}}_{c^{\mathtt{L}}} \geq \hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}.$$

Both f and $f^{\mathtt{L}}_{c^{\mathtt{L}}}$ are symmetric trade-off functions. For larger g, f becomes smaller (for all α).

Notice that the adversary gets better in hypothesis testing when f becomes smaller for larger g. Therefore, in the worst case, the total number $t = |d \setminus d'| + |d' \setminus d|$ of samples in which d and d' differ are distributed as $|d \setminus d'| = t$ with $|d' \setminus d| = 0$ (or vice versa) as this achieves the maximum value for g, i.e., $g = t$.

The difficulty of applying Theorems 4.5 and 4.6 is in computing the probability distribution $q_E(c)$ – the number of possible c may be exponential in g and E. Therefore, computing f may be prohibited and only the lower and upper bound functions can possibly be estimated. Here, we remark that in general the lower and upper bounds are not tight together for small E; only for larger E the probability distribution $q_E(c)$ will concentrate and lower and upper bounds exist that approach one another for large E.

In Appendices F and G we apply Theorems 4.5 and 4.6 to the DP analysis of subsampling and shuffling. Table 1 in the introduction summarizes the derived bounds[†].

- The stated lower bound $\approx G_{\sqrt{(e+\gamma)\min\{m,g\}gE}/\sigma}$ for individual clipping is $\leq G_{\sqrt{gE}/\sigma}$, the trade-off function which the adversary can approximately achieve and cannot further improve for batch clipping. This shows that, unless the lower bound is not tight and can be improved upon, batch clipping provides a better DP guarantee since $G_{\sqrt{gE}/\sigma}$ is closer to the ideal trade-off function $1 - \alpha$ (which corresponds to random guessing the hypothesis d or d').

- Shuffling appears to concentrate $q_E(c)$ more compared to subsampling. This leads to sharper bounds: For the cases that can be compared, the ranges provided for subsampling prove $G_{\sqrt{(1+1/\sqrt{2gE})gE}/\sigma}$-DP while for shuffling we can prove $G_{\sqrt{gE}/\sigma}$-DP which is closer to $1 - \alpha$ since $G_{\sqrt{(1+1/\sqrt{2gE})gE}/\sigma} \leq G_{\sqrt{gE}/\sigma}$. Here, we note that $G_{\sqrt{gE}/\sigma}$ lies in the reported ranges for subsampling and with improved techniques it may be able to show that $G_{\sqrt{gE}/\sigma}$-DP is also true for subsampling. For now, we can only show this for large E when $G_{\sqrt{(1+1/\sqrt{2gE})gE}/\sigma}$ tends to $G_{\sqrt{gE}/\sigma}$.

- The table shows upper and lower bounds that are functions of the number E of epochs, and are invariant with respect to the exact number of rounds per epoch. It turns out that even though more rounds lead to more updates that can be used for improved hypothesis testing, this is compensated for by each round update leaking less privacy due to the sampling strategy (subsampling or shuffling). So, we can freely choose the number of rounds in each epoch and this can be used to optimize for best accuracy.

- The group privacy analysis using the $\circ$ operator in the original f-DP framework [2] based on our $g = 1$ analysis gives a $G_{g\sqrt{E}/\sigma}$-DP like guarantee. This is not tight as our results show the much better tight $G_{\sqrt{gE}/\sigma}$-DP like dependency.

5 Experiments

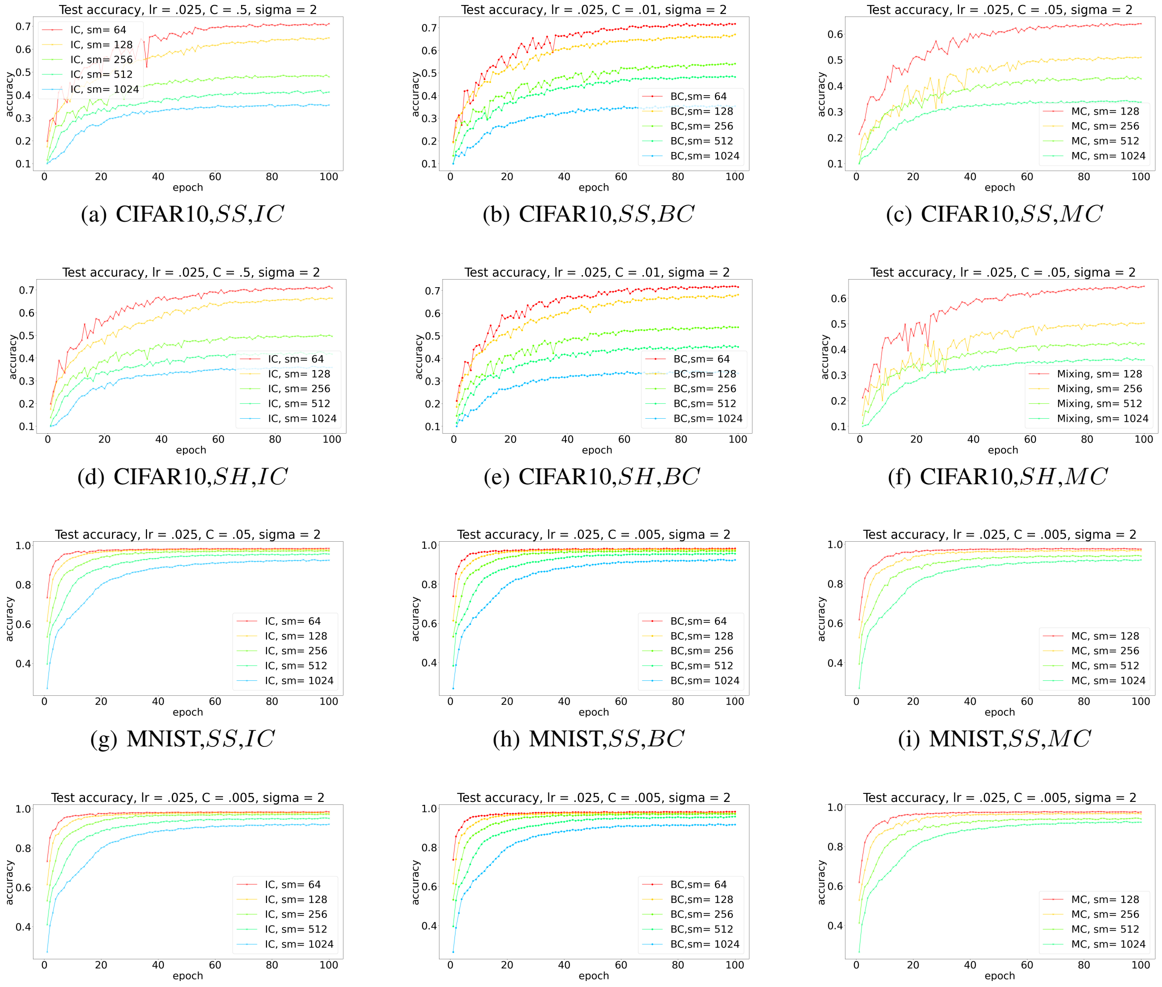
In this section we provide a study on extending the mini-batch SGD approach of DP-SGD to include batch clipping as is allowed by the DP analysis of our general framework. We demonstrate that Algorithm 1 can produce high accuracy models for different setups – in our experiments, batch clipping yields accuracies comparable to DP-SGD with individual clipping, see Table 2.

Table 2 with Figure 1 depict accuracies for CIFAR-10 [9] and MNIST [11] where we compute updates

$$U = \sum_{h=1}^{m} [\frac{1}{s} \sum_{i \in S_{b,h}} \nabla f(w, \xi_i)]_C \tag{6}$$

with $s = 1$ for Individual Clipping (IC) (this is DP-SGD), $m = 1$ for pure Batch Clipping (BC), and values for s and m in between these two extremes denoted by Mixed Clipping (MC). We implement mini-batch SGD and therefore we

[†] The table shows upper and lower bounds that are independent of N or generally hold for large N (as found in practice). Hence, the DP guarantees for $\mathcal{M}$ hold for all pairs of data sets d and d' with $\max\{|d \setminus d'|, |d' \setminus d|\} \leq g$ (for the g stated in the table). The condition $N = |d \cap d'| + g$ of the theorem can be discarded.



(a) CIFAR10,SS,IC (b) CIFAR10,SS,BC (c) CIFAR10,SS,MC

(d) CIFAR10,SH,IC (e) CIFAR10,SH,BC (f) CIFAR10,SH,MC

(g) MNIST,SS,IC (h) MNIST,SS,BC (i) MNIST,SS,MC

(j) MNIST,SH,IC (k) MNIST,SH,BC (l) MNIST,SH,MC

Figure 1: CIFAR10 and MNIST testing accuracy for batch size $|S_b| = sm = 64, 128, 256, 512, 1028$ where SS and SH denote SubSampling and SHuffling, and where IC, BC and MC denote Individual Clipping ($s = 1$), Batch Clipping ($m = 1$) and Mixed Clipping, respectively. For MC, we set $s = 64$ and $m = 2, 4, 8$ and 16. The used learning rate (step size) is $\mu = 0.025$ (denoted by 'lr'), the used noise is given by $\sigma = 2$ (denoted by 'sigma'), and the used clipping constant C is fine-tuned for each separate setting.

Sampling	Clipping	CIFAR-10	MNIST
SS	IC	71.09%/68.94%	98.41%/98.31%
SS	BC	71.7%/68.94%	98.21%/98.31%
SS	MC	63.97%/68.94%	97.5%/98.31%
SH	IC	70.77%/70.14%	98.38%/98.4%
SH	BC	71.52%/70.14%	98.33%/98.4%
SH	MC	64.68%/70.14%	96.97%/98.4%

Table 2: The best results among the different setups of Figure 1 for CIFAR-10 and MNIST with $\sigma = 2$ and learning rate $\eta = 0.025$ compared to mini-batch SGD without DP.

use the same w in our gradient computations in (6).[‡] The initial learning rate (step size) is $\mu = 0.025$ and, for each new epoch, the step size is decreased by a factor $\gamma = 0.9$ if there is a decrease in testing accuracy between the last two consecutive epochs. In our settings we use the same $\sigma = 2$ and compare experiments for each benchmark for the same $E = 100$ number of epochs gradient computations.

[‡] Notice that division by m is taken care of in lines 23 and 24 of Algorithm 1 and division by s is done in (6) itself.

For each different setup, we report the best result among our grid-search in Table 2. For the image classification model of MNIST, we use LeNet-5 [10] which yields 98.31% and 98.38% testing accuracy for mini-batch SGD without DP for subsampling and shuffling. For our demonstration of being able to achieve similar high accuracy models, we use the lightweight Convolutional Neural Network (CNN) in Table **??** for CIFAR10. This has both fast training and good accuracy; 68.94% and 70.14% for mini-batch SGD without DP for subsampling and shuffling.

Operation Layer	#Filters	Kernel size	Stride	Padding
$Conv2D$+$ReLu$	32	3×3	1×1	1×1
$AvgPool2d$	1	2×2	2×2	$-$
$Conv2D$+$ReLu$	64	3×3	1×1	1×1
$AvgPool2d$	1	2×2	2×2	$-$
$Conv2D$+$ReLu$	64	3×3	1×1	1×1
$AvgPool2d$	1	2×2	2×2	$-$
$Conv2D$+$ReLu$	128	3×3	1×1	1×1
$AdaptiveAvgPool2d$	$-$	$-$	$-$	$-$
$Softmax$	$-$	$-$	$-$	$-$

Table 3: CNN architecture for CIFAR-10 dataset

This section reports an initial study and future work is needed to explore the more general algorithmic freedom in (algorithm $\mathcal{A}$ of) our framework for improving accuracy in specific application scenarios. As a result of our experiments, we can already conclude that for mini-batch SGD batch clipping is preferred over individual clipping because it allows being implemented with less memory overhead. This is because the implementation of (6) with $m = 1$ can keep track of just the gradient of the sum in the forward backward algorithm since $\sum_{i \in S_b} \nabla_w f(w, \xi_i) = \nabla_w \sum_{i \in S_b} f(w, \xi_i)$, whereas mini-batch SGD with individual clipping needs to keep track of each of the gradients $\nabla_w f(w; \xi_i)$ since these need to be clipped before they can be summed together. This leads to a faster training process and smaller memory overhead compared to individual clipping, specifically, we trained the models using 11GB NVIDIA GTX1080ti GPUs with $15 - 30$ minutes running time for each batch clipping experiment and $4 - 6$ hours running time for each individual clipping experiment.

6 Conclusion

We have introduced a general algorithmic framework which allows strong f-DP guarantees in a slightly stronger adversarial model. (a) The resulting DP guarantees have simple tight closed form formulas that are easily evaluated and interpreted (no DP accountant is needed), contrary to existing f-DP guarantees (which use the more complex subsampling operator) in the adversarial model common in current literature. Existing f-DP guarantees have "the disadvantage that the expressions it yields are more unwieldy" [2]. (b) Our DP guarantees show invariance with respect to the number of rounds per epoch and show a $\sqrt{g}$ dependency for group privacy. (c) Our general framework allows batch clipping together with a wide variety of optimization algorithms. An open problem is whether advantages similar to (b) and (c) can be proven in the adversarial model common in current literature. And it also remains open into what extent the algorithmic freedom can be further exploited for improved accuracy and performance.

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 308–318. ACM, 2016.

[2] Jinshuo Dong, Aaron Roth, and Weijie Su. Gaussian differential privacy. *Journal of the Royal Statistical Society*, 2021.

[3] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[4] Cynthia Dwork. A firm foundation for private data analysis. *Communications of the ACM*, 54(1):86–95, 2011.

[5] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 486–503. Springer, 2006.

[6] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[7] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[8] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[9] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. CIFAR-10 (Canadian Institute for Advanced Research). URL `http://www.cs.toronto.edu/~kriz/cifar.html`.

[10] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

[11] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010. URL `http://yann.lecun.com/exdb/mnist/`.

[12] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data, 2017.

[13] Lam Nguyen, Phuong Ha Nguyen, Marten Dijk, Peter Richtárik, Katya Scheinberg, and Martin Takác. Sgd and hogwild! convergence without the bounded gradients assumption. In *International Conference on Machine Learning*, pages 3750–3758. PMLR, 2018.

[14] Lam M. Nguyen, Phuong Ha Nguyen, Peter Richtárik, Katya Scheinberg, Martin Takáč, and Marten van Dijk. New convergence aspects of stochastic gradient algorithms. *Journal of Machine Learning Research*, 20(176): 1–49, 2019.

[15] Nhuong Nguyen, Toan Nguyen, Phuong Ha Nguyen, Quoc Tran-Dinh, Lam Nguyen, and Marten Dijk. Hogwild! over distributed local data sets with linearly increasing mini-batch sizes. In *International Conference on Artificial Intelligence and Statistics*, pages 1207–1215. PMLR, 2021.

[16] Opacus. Opacus PyTorch library. Available from opacus.ai.

[17] Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147, 2013.

[18] Larry Wasserman and Shuheng Zhou. A statistical framework for differential privacy. *Journal of the American Statistical Association*, 105(489):375–389, 2010.

[19] Yuqing Zhu, Jinshuo Dong, and Yu-Xiang Wang. Optimal accounting of differential privacy via characteristic function. *arXiv preprint arXiv:2106.08567*, 2021.

[20] Fangyu Zou, Li Shen, Zequn Jie, Weizhong Zhang, and Wei Liu. A Sufficient Condition for Convergences of Adam and RMSProp. *arXiv preprint arXiv:1811.09358*, 2018.

A Gaussian Differential Privacy

Dong et al. [2] introduced the state-of-the-art DP formulation based on hypothesis testing. From the attacker's perspective, it is natural to formulate the problem of distinguishing two neighboring data sets d and d' based on the output of a DP mechanism $\mathcal{M}$ as a hypothesis testing problem:

$$H_0: \text{ the underlying data set is } d \quad \text{versus} \quad H_1: \text{ the underlying data set is } d'.$$

Here, neighboring means that either $|d \setminus d'| = 1$ or $|d' \setminus d| = 1$. More precisely, in the context of mechanism $\mathcal{M}$, $\mathcal{M}(d)$ and $\mathcal{M}(d')$ take as input representations r and r' of data sets d and d' which are 'neighbors.' The representations are mappings from a set of indices to data samples with the property that if $r(i) \in d \cap d'$ or $r'(i) \in d \cap d'$, then $r(i) = r'(i)$. This means that the mapping from indices to data samples in $d \cap d'$ is the same for the representation of d and the representation of d'. In other words the mapping from indices to data samples for d and d' only differ for indices corresponding to the differentiating data samples in $(d \setminus d') \cup (d' \setminus d)$. In this sense the two mappings (data set representations) are neighbors.

We define the Type I and Type II errors by

$$\alpha_\phi = \mathbb{E}_{o \sim \mathcal{M}(d)}[\phi(o)] \text{ and } \beta_\phi = 1 - \mathbb{E}_{o \sim \mathcal{M}(d')}[\phi(o)],$$

where ϕ in $[0, 1]$ denotes the rejection rule which takes the output of the DP mechanism as input. We flip a coin and reject the null hypothesis with probability ϕ. The optimal trade-off between Type I and Type II errors is given by the trade-off function

$$T(\mathcal{M}(d), \mathcal{M}(d'))(\alpha) = \inf_\phi \{\beta_\phi \ : \ \alpha_\phi \leq \alpha\},$$

for $\alpha \in [0, 1]$, where the infimum is taken over all measurable rejection rules ϕ. If the two hypotheses are fully indistinguishable, then this leads to the trade-off function $1 - \alpha$. We say a function $f \in [0, 1] \to [0, 1]$ is a trade-off function if and only if it is convex, continuous, non-increasing, and $0 \leq f(x) \leq 1 - x$ for $x \in [0, 1]$.

We define a mechanism $\mathcal{M}$ to be f-DP if f is a trade-off function and

$$T(\mathcal{M}(d), \mathcal{M}(d')) \geq f$$

for all neighboring d and d'. Proposition 2.5 in [2] is an adaptation of a result in [18] and states that a mechanism is (ϵ, δ)-DP if and only if the mechanism is $f_{\epsilon,\delta}$-DP, where

$$f_{\epsilon,\delta}(\alpha) = \min\{0, 1 - \delta - e^\epsilon \alpha, (1 - \delta - \alpha)e^{-\epsilon}\}.$$

We see that f-DP has the (ϵ, δ)-DP formulation as a special case. It turns out that the original DP-SGD algorithm can be tightly analysed by using f-DP.

A.1 Gaussian DP

In order to proceed, [2] first defines Gaussian DP as another special case of f-DP as follows: We define the trade-off function

$$G_\mu(\alpha) = T(\mathcal{N}(0, 1), \mathcal{N}(\mu, 1))(\alpha) = \Phi(\Phi^{-1}(1 - \alpha) - \mu),$$

where Φ is the standard normal cumulative distribution of $\mathcal{N}(0, 1)$. We define a mechanism to be μ-Gaussian DP if it is G_μ-DP. Corollary 2.13 in [2] shows that a mechanism is μ-Gaussian DP if and only if it is $(\epsilon, \delta(\epsilon))$-DP for all $\epsilon \geq 0$, where

$$\delta(\epsilon) = \Phi(-\frac{\epsilon}{\mu} + \frac{\mu}{2}) - e^\epsilon \Phi(-\frac{\epsilon}{\mu} - \frac{\mu}{2}). \tag{7}$$

Suppose that a mechanism $\mathcal{M}(d)$ computes some function $u(d) \in \mathbb{R}^n$ and adds Gaussian noise $\mathcal{N}(0, (c\sigma)^2\mathbf{I})$, that is, the mechanism outputs $o \sim u(d) + \mathcal{N}(0, (c\sigma)^2\mathbf{I})$. Suppose that c denotes the sensitivity of function $u(\cdot)$, that is,

$$\|u(d) - u(d')\| \leq c$$

for neighboring d and d'; the mechanism corresponding to one round update in Algorithm 1 has *sensitivity* $c = 2C$. After projecting the observed o onto the line that connects $u(d)$ and $u(d')$ and after normalizing by dividing by c, we have that differentiating whether o corresponds to d or d' is in the best case for the adversary (i.e., $\|u(d) - u(d')\| = c$) equivalent to differentiating whether a received output is from $\mathcal{N}(0, \sigma^2)$ or from $\mathcal{N}(1, \sigma^2)$. Or, equivalently, from $\mathcal{N}(0, 1)$ or from $\mathcal{N}(1/\sigma, 1)$. This is how the Gaussian trade-off function $G_{\sigma^{-1}}$ comes into the picture.

A.2 Subsampling

Besides implementing Gaussian noise, DP-SGD also uses sub-sampling: For a data set d of N samples, $\texttt{Sample}_m(d)$ selects a subset of size m from d uniformly at random. We define convex combinations

$$f_p(\alpha) = pf(\alpha) + (1-p)(1-\alpha)$$

with corresponding p-sampling operator

$$C_p(f) = \min\{f_p, f_p^{-1}\}^{**},$$

where the conjugate h^* of a function h is defined as

$$h^*(y) = \sup_x \{yx - h(x)\}$$

and the inverse h^{-1} of a trade-off function h is defined as

$$h^{-1}(\alpha) = \inf\{t \in [0,1] \mid h(t) \leq \alpha\} \tag{8}$$

and is itself a trade-off function (as an example, we notice that $G_\mu = G_\mu^{-1}$ and we say G_μ is symmetric). Theorem 4.2 in [2] shows that if a mechanism $\mathcal{M}$ on data sets of size N is f-DP, then the subsampled mechanism $\mathcal{M} \circ \texttt{Sample}_m$ is $C_{m/N}(f)$-DP.

The intuition behind operator C_p is as follows. First, $\texttt{Sample}_m(d)$ samples the differentiating element between d and d' with probability p. In this case the computations $\mathcal{M} \circ \texttt{Sample}_m(d)$ and $\mathcal{M} \circ \texttt{Sample}_m(d')$ are different and hypothesis testing is possible with trade-off function $f(\alpha)$. With probability $1-p$ no hypothesis testing is possible and we have trade-off function $1-\alpha$. This leads to the convex combination f_p.

Second, we notice if $h = T(\mathcal{M}(d), \mathcal{M}(d'))$, then $h^{-1} = T(\mathcal{M}(d'), \mathcal{M}(d))$. Therefore, if $\mathcal{M}$ is f-DP (which holds for all pairs of neighboring data sets, in particular, for the pairs (d, d') and (d', d)), then both $h \geq f$ and $h^{-1} \geq f$ and we have a symmetric upper bound $\min\{h, h^{-1}\} \geq f$. Since f is a trade-off function, f is convex and we can compute a tighter upper bound: f is at most the largest convex function $\leq \min\{h, h^{-1}\}$, which is equal to the double conjugate $\min\{h, h^{-1}\}^{**}$. From this we obtain the definition of operator C_p.

A.3 Composition

The tensor product $f \otimes h$ for trade-off functions $f = T(P, Q)$ and $h = T(P', Q')$ is well-defined by

$$f \otimes h = T(P \times P', Q \times Q').$$

Let $y_i \leftarrow \mathcal{M}_i(\texttt{aux}, d)$ with $\texttt{aux} = (y_1, \ldots, y_{i-1})$. Theorem 3.2 in [2] shows that if $\mathcal{M}_i(\texttt{aux}, .)$ is f_i-DP for all $\texttt{aux}$, then the composed mechanism $\mathcal{M}$, which applies $\mathcal{M}_i$ in sequential order from $i=1$ to $i=T$, is $(f_1 \otimes \ldots \otimes f_T)$-DP. The tensor product is commutative.

As a special case Corollary 3.3 in [2] states that composition of multiple Gaussian operators G_{μ_i} results in G_μ where

$$\mu = \sqrt{\sum_i \mu_i^2}.$$

A.4 Tight Analysis DP-SGD

We are now able to formulate the differential privacy guarantee of original DP-SGD since it is a composition of subsampled Gaussian DP mechanisms. Theorem 5.1 in [2] states that DP-SGD as introduced in [1] is

$$C_{m/N}(G_{\sigma^{-1}})^{\otimes T}\text{-DP},$$

where $T = (N/m) \cdot E$ is the total number of local rounds. Since each of the theorems and results from [2] enumerated above are exact, we have a tight analysis. This leads in [19] to a (tight) differential privacy accountant (using complex characteristic functions for each of the two hypotheses based on taking Fourier transforms), which can be used by a client to keep track of its current DP guarantee and to understand when to stop helping the server to learn a global model. Because the accountant is tight, it improves over the momentum accountant method of [1].

A.5 Group Privacy

Theorem 2.14 in [2] analyzes how privacy degrades if d and d' do not differ in just one sample, but differ in g samples. If a mechanism is f-DP, then it is

$$[1-(1-f)^{\circ g}]\text{-DP}$$

for groups of size g (where $\circ g$ denotes the g-fold iterative composition of function $1-f$, where 1 denotes the constant integer value 1 and not the identity function, i.e., $(1-f)(\alpha)=1-f(\alpha)$). This is a tight statement in that *there exist* f such that the trade-off function for groups of size g cannot be bounded better. In particular, for $f=G_\mu$ we have $G_{g\mu}$-DP for groups of size g.

The intuition behind the $[1-(1-f)^{\circ g}]$-DP result is that the adversary can create a sequence of data sets $d_0=d,\, d_1,\, \ldots,\, d_{g-1},\, d_g=d'$ such that each two consecutive data sets d_i and d_{i+1} are neighboring. We know that $T(\mathcal{M}(d_i),\mathcal{M}(d_{i+1}))\geq f$. For each rejection rule we may plot a point (in x and y coordinates)

$$(\mathbb{E}_{o\sim\mathcal{M}(d_i)}[\phi(o)],\ \mathbb{E}_{o\sim\mathcal{M}(d_{i+1})}[\phi(o)]).$$

Since $f(\alpha)$ is a lower bound on the Type I vs Type II error curve, the resulting collection of points is upper bounded by the curve $1-f(\alpha)$. We have that $\alpha=\mathbb{E}_{o\sim\mathcal{M}(d_i)}[\phi(o)]$ is mapped to

$$\mathbb{E}_{o\sim\mathcal{M}(d_{i+1})}[\phi(o)]\leq 1-f(\alpha)=(1-f)(\alpha).$$

By transitivity, we have that $\alpha=\mathbb{E}_{o\sim\mathcal{M}(d=d_0)}[\phi(o)]$ is mapped to

$$\mathbb{E}_{o\sim\mathcal{M}(d'=d_g)}[\phi(o)]\leq(1-f)^{\circ g}(\alpha).$$

This yields the lower bound

$$T(\mathcal{M}(d),\mathcal{M}(d'))\geq 1-(1-f)^{\circ g}$$

on the Type I vs Type II error curve.

Let $\phi[\alpha]$ denote a rejection rule that realizes the mapping from

$$\alpha=\mathbb{E}_{o\sim\mathcal{M}(d_i)}[\phi[\alpha](o)]\ \text{ to }\ (1-f)(\alpha)=\mathbb{E}_{o\sim\mathcal{M}(d_{i+1})}[\phi[\alpha](o)].$$

Then the mapping from $(1-f)^{\circ i}(\alpha)=\mathbb{E}_{o\sim\mathcal{M}(d_i)}[\phi(o)]$ to $(1-f)^{\circ(i+1)}(\alpha)=\mathbb{E}_{o\sim\mathcal{M}(d_{i+1})}[\phi(o)]$ is realized by $\phi=\phi[(1-f)^{\circ i}(\alpha)]$. This shows that the lower bound $1-(1-f)^{\circ g}$ is tight only if we can choose all $\phi[(1-f)^{\circ i}(\alpha)]$ equal to one another. This is not the case for DP-SGD for which it turns out that this lower bound is not tight at all; rather than a multiplicative factor g as in the mentioned $G_{g\mu}$-DP guarantee we see a $\sqrt{g}$ dependency for adversary $\mathcal{A}_1$ in this paper (and this should also hold for the seemingly weaker adversary $\mathcal{A}_0$). This is done by considering how, due to sub-sampling, the g differentiating samples are distributed across all the rounds within an epoch and how composition of trade-off functions across rounds yields the $\sqrt{g}$ dependency.

B Strong Adversarial Model

We assume an adversary who knows the differentiating samples in $d\setminus d'$ and $d'\setminus d$, but who a-priori (before mechanism $\mathcal{M}$ is executed) may only know (besides say a 99% characterization of $d\cap d'$) an estimate of the number of samples in the intersection of d and d', i.e., the adversary knows $|d\cap d'|+noise$ where the noise is large enough to yield a 'sufficiently strong' DP guarantee with respect to the size of the used data set (d or d'). Since $\mathcal{M}$ does not directly reveal the size of the used data set, we assume (as in prior literature) that the effect of $N=|d|\neq N'=|d'|$ contributes at most a very small amount of privacy leakage, sufficiently small to be discarded in our DP analysis: That is, we may as well assume $N=N'$ in our DP analysis.

In this setting of $N=N'$ the DP analysis in prior work considers an adversary who can mimic mechanism $\mathcal{M}\circ\mathsf{Sample}_m$ in that it can replay into large extent how Sample_m samples the used data set (d or d'): We say a round has k differentiating data samples if Sample_m sampled a subset of indices which contains exactly k indices of differentiating data samples from $(d\setminus d')\cup(d'\setminus d)$. The adversary knows how Sample_m operates and can derive a joint probability distribution $\mathbb{P}$ of the number of differentiating data samples for each round within the sequence of rounds that define the series of epochs during which updates are computed. We consider two types of strong adversaries in our proofs when bounding trade-off functions:

Adversary $\mathcal{A}_0$ does not know the exact instance drawn from $\mathbb{P}$ but who is, in the DP proof, given the ability to realize for each round the trade-off function $f_k(\alpha)$ that corresponds to hypothesis testing between $\mathcal{M}\circ\mathtt{Sample}_m(d)$ and $\mathcal{M}\circ\mathtt{Sample}_m(d')$ if $\mathtt{Sample}_m$ has selected k differentiating samples in that round. Adversary $\mathcal{A}_0$ in the DP analysis that characterizes $f_k(\alpha)$ is given knowledge about the mapping from indices to values in d or d'. Here (as discussed before), the mapping from indices to values in $d \cap d'$ is the same for the mapping from indices to values in d and the mapping from indices to values in d'. Furthermore, the adversary can replay how Sample_m samples a subset of m indices from[§] $\{1, \ldots, N = N'\}$, and it knows all the randomness used by $\mathcal{M}$ before $\mathcal{M}$ adds Gaussian noise for differential privacy (this includes when and how the interrupt service routine overwrites the local model). This strong adversary represents a worst-case scenario for the 'defender' when analyzing the differential privacy of a single round. For DP-SGD this analysis for neighboring data sets leads to the argument of Section A.2 where with probability p (i.e., $k = 1$) the adversary can achieve trade-off function $f(\alpha)$ and with probability $1 - p$ (i.e., $k = 0$) can achieve trade-off function $1 - \alpha$ leading ultimately to operator C_p. This in turn leads to the trade-off function $C_{m/N}(G_{\sigma^{-1}})^{\otimes T}$ with $p = m/N$, which is *tight for adversary* $\mathcal{A}_0$. We notice that adversary $\mathcal{A}_0$ is used in DP analysis of current literature including the moment accountant method of [1] for analysing (ϵ, δ)-DP and analysis of divergence based DP measures.

In the DP analysis adversary $\mathcal{A}_0$ is given knowledge about the number k of differentiating samples when analysing a single round. That is, it is given an instance of $\mathbb{P}$ projected on a single round. We notice that in expectation the sensitivity (see Section A.1) of a single round as observed by adversary $\mathcal{A}_0$ for neighboring data sets is equal to $(1 - p) \cdot 0 + p \cdot 2C = (m/N) \cdot 2C$ and this gives rise to an 'expected' trade-off function $G_{1/(\sigma N/m)}$. Composition over $c^2(N/m)^2$ rounds gives $G_{1/\sigma}$. This leads us to believe that $C_{m/N}(G_{\sigma^{-1}})^{\otimes T}$ converges to $G_{c \cdot h(\sigma)}$ for $T = c^2(N/m)^2 \to \infty$ (or, equivalently, $\sqrt{T} \cdot m/N = c$ with $T \to \infty$ and $N \to \infty$) where $h(\sigma)$ is some function that only depends on σ. This intuition is confirmed by Corollary 5.4 in [2].

We define the second type of adversary $\mathcal{A}_1$ as one who has knowledge about a full instance of $\mathbb{P}$, not just a projection of $\mathbb{P}$ on a single round as for adversary $\mathcal{A}_0$. This allows a DP analysis that into some extent computes a convex combination of trade-off functions that each characterize all the rounds together as described by an instance of $\mathbb{P}$. This gives adversary $\mathcal{A}_1$ more information and the resulting DP guarantees should be weaker compared to the analysis based on adversary $\mathcal{A}_0$ (because adversary $\mathcal{A}_1$ considers a more worst-case leakage scenario). Since each epoch has N/m rounds and since m/N is equal to the probability that $\mathtt{Sample}_m$ selects a differentiating sample in a round when considering neighboring data sets, we have that a single epoch of N/m rounds has in expectation exactly one round with one differentiating sample while all other rounds only use non-differentiating samples. This is a composition of $G_{\sigma^{-1}}$ with $N/m - 1$ times G_0. We have that the trade-off function for $c^2 \cdot (N/m)$ rounds has in expectation a trade-off function $G_{c/\sigma}$ (this is confirmed by the $G_{\sqrt{E}/\sigma}$-DP guarantees in Table 1 where the total number of gradient computations is equal to $c^2 \cdot (N/m)$ times m, which is equal to $E = c^2$ epochs of size N). This shows convergence to some $G_{c \cdot h(\sigma)}$ for $T = c^2 \cdot (N/m) \to \infty$ rounds. This is indeed a weaker statement compared to the one for adversary $\mathcal{A}_0$.

C Helpful Lemmas

Our DP analysis depends on the next lemmas which we state first. Let a mechanism $\mathcal{M}$ be defined as a probabilistic sum of base mechanisms $\mathcal{M}_i$, that is,

$$\mathcal{M} = \sum_i q_i \mathcal{M}_i$$

for some probabilities q_i with $\sum_i q_i = 1$. $\mathcal{M}$ executes mechanism $\mathcal{M}_i$ with probability q_i. Let f_i be a trade-off function for $\mathcal{M}_i$, that is,

$$T(\mathcal{M}_i(d), \mathcal{M}_i(d')) \geq f_i.$$

Then the following lemmas provide bounds on the trade-off function $T(\mathcal{M}(d), \mathcal{M}(d'))$.

Lemma C.1. *Let $T(\mathcal{M}_i(d), \mathcal{M}_i(d')) \geq f_i$ and define*

$$f(\alpha) = \inf_{\{\alpha_i\}} \left\{ \sum_i q_i f_i(\alpha_i) \;\middle|\; \sum_i q_i \alpha_i = \alpha \right\}.$$

Then f is a trade-off function and $T(\mathcal{M}(d), \mathcal{M}(d')) \geq f$.

In addition, if all f_i are symmetric (i.e., f_i^{-1} as defined in (8) is equal to f_i), then f is symmetric as well. In this case we also have $f_i(f_i(\alpha)) = \alpha$ and $f(f(\alpha)) = \alpha$ for $\alpha \in [0, 1]$.

Lemma C.2. *Suppose that $f_1 \leq f_2 \leq f_3 \leq \ldots$ with $T(\mathcal{M}_i(d), \mathcal{M}_i(d')) \geq f_i$. Let $p_t = \sum_{i<t} q_i$. Then, we have the lower bound $T(\mathcal{M}(d), \mathcal{M}(d'))(\alpha) \geq f(\alpha) \geq f_t(\min\{1, \alpha + p_t\})$, where f is as defined in Lemma C.1.*

[§] By assuming $N = N'$ in the DP analysis, knowledge of how Sample_m samples a subset of indices cannot be used to differentiate the hypotheses of d versus d' based on their sizes (since the index set corresponding to d is exactly the same as the index set corresponding to d').

Lemma C.3. *Suppose that $f_1 \leq f_2 \leq f_3 \leq \ldots$ with $f_i = T(\mathcal{M}_i(d), \mathcal{M}_i(d'))$. Let $p_t = \sum_{i>t} q_i$. Assume there exists a probability distribution P on the real numbers with log concave probability density and there exist real numbers $\{t_i\}$ such that $f_i = T(P, t_i + P)$. Then, we have the upper bound $p_t + (1-p_t)f_t(\min\{1, \alpha/(1-p_t)\}) \geq T(\mathcal{M}(d), \mathcal{M}(d'))(\alpha)$.*

Lemma C.4. *If $f = f_i = T(\mathcal{M}_i(d), \mathcal{M}_i(d'))$ for all i, then $T(\mathcal{M}(d), \mathcal{M}(d')) = f$.*

The next lemma allows us to solve for the infinum in Lemma C.1.

Lemma C.5. *Let*

$$f(\alpha) = \inf_{\{\alpha_i\}:\sum_i q_i\alpha_i = \alpha} \sum_i q_i f_i(\alpha_i)$$

be the trade-off function of Lemma C.1 where $f_i = G_{\mu_i}$. Let $\Lambda(\alpha)$, $\alpha \in [0,1]$, be implicitly defined by the equation

$$1 - \alpha = \sum_i q_i \Phi(\frac{\Lambda(\alpha)}{\mu_i} + \frac{\mu_i}{2}).$$

Then,

$$f(\alpha) = \sum_i q_i \Phi(\frac{\Lambda(\alpha)}{\mu_i} - \frac{\mu_i}{2}).$$

C.1 Proof of Lemma C.1

We want to prove a lower bound of the trade-off function of $\mathcal{M} = \sum_i q_i \mathcal{M}_i$ in terms of trade-off functions f_i for $\mathcal{M}_i$: We define

$$\alpha_{i,\phi} = \mathbb{E}_{o\sim\mathcal{M}_i(d)}[\phi(o)] \text{ and } \beta_{i,\phi} = 1 - \mathbb{E}_{o\sim\mathcal{M}_i(d')}[\phi(o)]$$

and notice that for $\mathcal{M}$ we have

$$\alpha_\phi = \sum_i q_i \alpha_{i,\phi} \text{ and } \beta_\phi = \sum_i q_i \beta_{i,\phi}.$$

We derive

$$\begin{aligned} T(\mathcal{M}(d), \mathcal{M}(d'))(\alpha) &= \inf_\phi \{\beta_\phi \ : \ \alpha_\phi \leq \alpha\} \\ &= \inf_\phi \{\sum_i q_i \beta_{i,\phi} \ : \ \sum_i q_i \alpha_{i,\phi} \leq \alpha\} \\ &= \inf_{\{\alpha_i\}:\sum_i q_i\alpha_i = \alpha} \inf_\phi \{\sum_i q_i \beta_{i,\phi} \ : \ \forall_i \alpha_{i,\phi} \leq \alpha_i\} \qquad (9) \\ &\geq \inf_{\{\alpha_i\}:\sum_i q_i\alpha_i = \alpha} \sum_i q_i \inf_\phi \{\beta_{i,\phi} \ : \ \alpha_{i,\phi} \leq \alpha_i\} \\ &\geq \inf_{\{\alpha_i\}:\sum_i q_i\alpha_i = \alpha} \sum_i q_i f_i(\alpha_i). \end{aligned}$$

Notice that the lower bound is indeed a trade-off function since it is continuous, non-increasing, at least 0, and $\leq 1-\alpha$ for $\alpha \in [0,1]$, and convexity follows from the next argument: Let $\alpha = p\alpha_0 + (1-p)\alpha_1$. By convexity of f_i, $f_i(p\alpha_{0,i} + (1-p)\alpha_{1,i}) \leq pf_i(\alpha_{0,i}) + (1-p)f_i(\alpha_{1,i})$. Convexity of the lower bound follows from

$$\begin{aligned} & \inf_{\{\alpha_i\}:\sum_i q_i\alpha_i = p\alpha_0 + (1-p)\alpha_1} \sum_i q_i f_i(\alpha_i) \\ = & \inf_{\{\alpha_i = p\alpha_{0,i} + (1-p)\alpha_{1,i}\}:\sum_i q_i\alpha_{0,i} = \alpha_0, \sum_i q_i\alpha_{1,i} = \alpha_1} \sum_i q_i f_i(\alpha_i) \\ \leq & \inf_{\{\alpha_i = p\alpha_{0,i} + (1-p)\alpha_{1,i}\}:\sum_i q_i\alpha_{0,i} = \alpha_0, \sum_i q_i\alpha_{1,i} = \alpha_1} p\sum_i q_i f_i(\alpha_{0,i}) + (1-p)\sum_i q_i f_i(\alpha_{1,i}) \\ = & \ p \inf_{\{\alpha_{0,i}\}:\sum_i q_i\alpha_{0,i} = \alpha_0} \sum_i q_i f_i(\alpha_{0,i}) + (1-p) \inf_{\{\alpha_{1,i}\}:\sum_i q_i\alpha_{1,i} = \alpha_1} \sum_i q_i f_i(\alpha_{1,i}). \end{aligned}$$

In addition, if all f_i are symmetric, then

$$\begin{aligned} f^{-1}(\alpha) &= \inf\{t \in [0,1] \mid f(t) \leq \alpha\} \\ &= \inf\{t \in [0,1] \mid \inf\{\sum_i q_i f_i(\alpha_i) \mid \sum_i q_i\alpha_i = t\} \leq \alpha\}. \end{aligned}$$

Since all f_i are trade-off functions and therefore non-increasing, increasing $t = \sum_i q_i \alpha_i$ implies that each α_i in the sum $\sum_i q_i f_i(\alpha_i)$ can be increased, hence, $\sum_i q_i f_i(\alpha_i)$ can be decreased. This implies that the infinum of $\sum_i q_i f_i(\alpha_i)$ will be smaller. We want the smallest (infinum) t such that the infinum of $\sum_i q_i f_i(\alpha_i)$ is at most α. So, we want the smallest (infinum) $t = \sum_i q_i \alpha_i$ for which $\sum_i q_i f_i(\alpha_i) = \alpha$. This yields

$$\begin{aligned}
f^{-1}(\alpha) &= \inf\{t \in [0,1] \mid \inf\{\sum_i q_i f_i(\alpha_i) \mid \sum_i q_i \alpha_i = t\} \leq \alpha\} \\
&= \inf\{\sum_i q_i \alpha_i \mid \sum_i q_i f_i(\alpha_i) = \alpha\}.
\end{aligned}$$

Since all f_i are symmetric, we have

$$f_i(\alpha) = f_i^{-1}(\alpha) = \inf\{t \in [0,1] \mid f(t) \leq \alpha\}.$$

Since $f_i(t)$ is non-increasing, this implies that, for $\alpha \in [0,1]$, $f_i(\alpha) = t$ for some $t \in [0,1]$ such that $f_i(t) = \alpha$. This proves $f_i(f_i(\alpha)) = \alpha$. And by substituting $\beta_i = f_i(\alpha_i)$ we have

$$\begin{aligned}
f^{-1}(\alpha) &= \inf\{\sum_i q_i \alpha_i \mid \sum_i q_i f_i(\alpha_i) = \alpha\} \\
&= \inf\{\sum_i q_i f_i(f_i(\alpha_i)) \mid \sum_i q_i f_i(\alpha_i) = \alpha\} \\
&= \inf\{\sum_i q_i f_i(\beta_i) \mid \sum_i q_i \beta_i = \alpha\} = f(\alpha).
\end{aligned}$$

We conclude that f is symmetric.

The above derivation for $f_i(f_i(\alpha)) = \alpha$ also holds for f and we have that $f(f(\alpha)) = \alpha$ for $\alpha \in [0,1]$.

C.2 Proof of Lemma C.2

The lower bound follows from Lemma C.1 with

$$\begin{aligned}
\sum_i q_i f_i(\alpha_i) &\geq \sum_{i \geq t} q_i f_i(\alpha_i) \geq \sum_{i \geq t} q_i f_t(\alpha_i) = \sum_{i \geq t} q_i f_t(\alpha_i) + \sum_{i<t} q_i f_t(1) \\
&\geq f_t(\sum_{i \geq t} q_i \alpha_i + \sum_{i<t} q_i) = f_t(\alpha + \sum_{i<t} q_i(1-\alpha_i)) \geq f_t(\min\{1, \alpha + p_t\}).
\end{aligned}$$

C.3 Proof of Lemma C.3

We refer to Proposition A.3 and the proof of (6) in [2] from which we conclude that, for all $\hat{\alpha} \in [0,1]$, there exists a rejection rule ϕ^* for which $\hat{\alpha} = \mathbf{E}_{o \sim P}[\phi^*(o)]$ and, for all i,

$$\beta_{i,\phi^*} = \inf_{\phi}\{\beta_{i,\phi} : \alpha_{i,\phi} \leq \hat{\alpha}\}.$$

For completeness, it turns out that the rejection rule ϕ^* is defined by a threshold mechanism: Reject a sample o if $o \geq t$ with $t = F^{-1}(1-\hat{\alpha})$ where F is the cdf of probability distribution P.

The upper bound follows from (9) by choosing $\alpha_i = 0$ for $i > t$, $\alpha_i = \min\{1, \alpha/(1-p_t)\}$ for $i \leq t$ with $p_t = \sum_{i>t} q_i$, and choose the rejection rule ϕ^* corresponding to $\hat{\alpha} = \min\{1, \alpha/(1-p_t)\}$. We derive

$$\begin{aligned}
\inf_\phi \{\sum_i q_i \beta_{i,\phi} \ : \ \forall_i \alpha_{i,\phi} \leq \alpha_i\} &\leq \inf_\phi \{\sum_{i>t} q_i + \sum_{i\leq t} q_i \beta_{i,\phi} \ : \ \forall_i \alpha_{i,\phi} \leq \alpha_i\} \\
&= p_t + \inf_\phi \{\sum_{i\leq t} q_i \beta_{i,\phi} \ : \ \forall_i \alpha_{i,\phi} \leq \alpha_i\} \\
&= p_t + \inf_\phi \{\sum_{i\leq t} q_i \beta_{i,\phi} \ : \ \forall_i \alpha_{i,\phi} \leq \min\{1, \alpha/(1-p_t)\}\} \\
&\leq p_t + \sum_{i\leq t} q_i \beta_{i,\phi^*} \\
&= p_t + \sum_{i\leq t} q_i f_i(\min\{1, \alpha/(1-p_t)\}) \\
&\leq p_t + \sum_{i\leq t} q_i f_t(\min\{1, \alpha/(1-p_t)\}) \\
&= p_t + (1-p_t) f_t(\min\{1, \alpha/(1-p_t)\}).
\end{aligned}$$

C.4 Proof of Lemma C.4

The lemma is about the special case where $f = f_i$ for all i. By taking $t = 1$ in the lower bound of the second statement, we have $p_t = \sum_{i<t} q_i = 0$ and lower bound $f_t(\alpha) = f(\alpha)$. By taking t equal to the largest index in the upper bound of the third statement, we have $p_t = \sum_{i>t} q_i = 0$ and upper bound $f_t(\alpha) = f(\alpha)$ (for the upper bound in this special case we do not need to satisfy $f_i = T(P, t_i + P)$). Combination of these bounds yields $T(\mathcal{M}(d), \mathcal{M}(d')) = f$.

C.5 Proof of Lemma C.5

Proof: The infinum can be solved by using a Lagrange multiplier: We define

$$\sum_i q_i G_{\mu_i}(\alpha_i) - \lambda(\alpha - \sum_i q_i \alpha_i)$$

as a function of all α_i and λ and find its stationary points. We remind the reader that $G_\mu(x) = \Phi(\Phi^{-1}(1-x) - \mu)$ and $\Phi'(x) = e^{-x^2/2}/\sqrt{2\pi}$, hence,

$$\begin{aligned}
G'_\mu(x) &= \Phi'(\Phi^{-1}(1-x) - \mu) \cdot \frac{1}{\Phi'(\Phi^{-1}(1-x))} \cdot (-1) \\
&= -\frac{e^{-(\Phi^{-1}(1-x)-\mu)^2/2}}{e^{-\Phi^{-1}(1-x)^2/2}} = -e^{\mu\Phi^{-1}(1-x)-\mu^2/2}. \qquad (10)
\end{aligned}$$

For the stationary point we have the equations

$$\begin{aligned}
0 &= -e^{\mu_i \Phi^{-1}(1-\alpha_i) - \mu_i^2/2} + \lambda, \\
\alpha &= \sum_i q_i \alpha_i.
\end{aligned}$$

This shows that

$$\alpha_i = 1 - \Phi(\frac{\ln(\lambda)}{\mu_i} + \frac{\mu_i}{2})$$

and therefore

$$1 - \alpha = \sum_i q_i \Phi(\frac{\ln(\lambda)}{\mu_i} + \frac{\mu_i}{2}).$$

The last equation can be used to solve for $\ln(\lambda)$ (this is possible because $\ln(\lambda) \to -\infty$ makes the sum zero, while $\ln(\lambda) \to +\infty$ makes the sum one) with which the different α_i can be computed using the first equation, which in turn allows us to evaluate the trade-off function for the overall mechanism in α. Notice that

$$G_{\mu_j}(\alpha_j) = \Phi(\frac{\ln(\lambda)}{\mu_j} - \frac{\mu_j}{2}).$$

Let $\ln(\lambda)$ be represented as the function value $\Lambda(\alpha)$ and the lemma follows.

D Proof DP Analyis

An adversary, who observes a transmitted noised local update $\bar{U}$, wants to gain information about whether training set d or d' was locally used by the client. We analyse the general situation where sets d and d' differ in a subset of samples. More precisely,

$$d = (d \cap d') + z \text{ and } d' = (d \cap d') + z'$$

for some sets z and z'. Sets z and z' contain differentiating samples that can be used by the adversary to conclude whether d or d' was used in the computation of $\bar{U}$.

D.1 Simulator

Suppose that d is the truth. Let U correspond to a computation based on a subset $\{\xi_i \in d : i \in S_b\}$. The adversary observes $\bar{U} \leftarrow U + \mathcal{N}(0, (2C\sigma)^2\mathbf{I})$. If the used $\{\xi_i \in d : i \in S_b\}$ is a subset of the intersection $d \cap d'$, then none of the samples ξ_i are differentiating. For each ξ_i there exists a $\xi'_j \in d'$ such that $\xi_i = \xi'_j$. In other words, there exists a simulator with access to d' (but not d) who could have produced U (with equal probability). This means that U and as a consequence $\bar{U}$ cannot be used by the adversary to differentiate whether d or d' was used.

Now suppose that $\{\xi_i \in d : i \in S_b\}$ has some differentiating samples in z. Now computation of U can be simulated by a simulator with access to d' plus the differentiating samples in z. This is because each ξ_i is either in $d \cap d' \subseteq d'$ or in z. Suppose that $\{\xi_i \in d : i \in S_b\}$ contains exactly t differentiating samples from z.

If d' were the truth, then the algorithm that produces the local update has no access to z. At best it can mimic the simulator for the non-differentiating $\xi_i \in d \cap d'$ and choose t other samples from d' to replace the t differentiating samples from z (according to some strategy). Let S'_b correspond to the set of samples $\{\xi'_i \in d' : i \in S'_b\}$ chosen by the mimicked simulator.

In a similar way, the mimicking of the simulator produces a partition $\{S'_{b,h}\}_{h=1}^{m}$. We have that $\{\xi_i : i \in S_{b,h}\}$ has exactly t_h differentiating samples if and only if $\{\xi_i : i \in S_{b,h}\}$ has exactly t_h differentiating samples. Their non-differentiating samples are the same. Notice that $\sum_{h=1}^{m} t_h = t$.

The same argument of the mimicking of the simulator also holds at the scale of an epoch which produces $\{S'_b\}_{b=1}^{N/(ms)}$.

The above argument is sound because $\texttt{Sample}_{s,m}$ samples from a set of indices $\{1, \ldots, N\}$ and does not sample from the data set d directly. This means that sampling is independent of the data set. And we can formulate our simulator by choosing a suitable index to data sample mapping (permutation). Since any permutation from indices to data samples is equally likely and randomized before each call to $\texttt{Sample}_{s,m}$, see Algorithm 1, the way the mimicked simulator samples from a set of size $N' = |d'|$ cannot be distinguished from how $\texttt{Sample}_{s,m}$ samples from a set of size $N = |d|$ if $N = N'$.

For the subsampling strategy, it matters only very little whether $\texttt{Sample}_{s,m}$ samples from a set of size N and the mimicked simulator samples from an index set of size $N' = |d'|$ with $N \neq N'$. This is because each set S_b is a random subset of size ms out of N indices where in practice $ms \ll N$ and N' is close to N (there are at most g differentiating samples), hence, the difference in frequency of selecting a specific sample by $\texttt{Sample}_{s,m}$ or by the mimicked simulator is very small, which we simply *assume leads to privacy leakage small enough to be discarded in our DP analysis* (previous work also builds on this assumption even though previous work does not state this assumption explicitly). Similarly, the shuffling strategy for $N \neq N'$ is assumed to suffer sufficiently small privacy leakage that can be discarded.

Of course, the client should not reveal the exact value of $N = |d|$ of his local data set d (as part of its mechanism), otherwise, the adversary can use this information to figure out whether, for example, d or d' with one extra data sample is being used. Notice that in practice, the adversary indeed does not exactly know N or N'. The adversary only knows the differentiating samples z and z' and a lot of information about the kind of data samples that are in the intersection $d \cap d'$ without knowing exactly its size. It is realistic to assume that, besides knowing the differentiating samples, the adversary only knows $|d \cap d'| + noise$ as a-priori information where the noise is large enough to yield a sufficiently strong DP guarantee in itself.

D.2 Hypothesis Testing

In order to distinguish whether the observed noised update corresponds to d or d', the best case for the adversary is to have knowledge about S_b and S'_b together with the index mappings to d and d' so that it can compute U corresponding to S_b and compute U' corresponding to S'_b. In order to be able to do so, we also give the adversary access to the randomness used by algorithm $\mathcal{A}$ (this includes when and how the local model w is overwritten by the interrupt service routine). This means that the adversary needs to perform hypothesis testing on whether the observed noised update comes from $U + \mathcal{N}(0, (2C\sigma)^2\mathbf{I})$ or $U' + \mathcal{N}(0, (2C\sigma)^2\mathbf{I})$.

D.3 Adversarial Model

Above defines a strong adversary $\mathcal{A}_1$ as discussed separately in Appendix B. In practice, the adversary does not know S_b or S'_b and we may treat U and U' as random variables rather than fixed vectors. There may be a plausible assumption on how data samples are generated such that the probability distribution for U and U' can be characterized. If there exists a (round dependent) distribution $\hat{\mathcal{N}}$ centered around 0 such that $U \leftarrow u + \hat{\mathcal{N}}$ and $U' \leftarrow u' + \hat{\mathcal{N}}$ for fixed vectors u and u' that can be computed by the adversary, then the adversary needs to do hypothesis testing between $u + \hat{\mathcal{N}} + \mathcal{N}(0, (2C\sigma)^2)$ and $u' + \hat{\mathcal{N}} + \mathcal{N}(0, (2C\sigma)^2)$. This reduces privacy leakage since hypothesis testing becomes less accurate because of the uncertainty added by $\hat{\mathcal{N}}$. Our strong (worst-case) adversary has no such uncertainty.

D.4 Gaussian Trade-Off Function

Let $\texttt{rand}$ denote the used randomness by $\mathcal{A}$. Then, given our strong adversarial model, we may write $a_h = \mathcal{A}(\texttt{rand}; w, \{\xi_i\}_{i \in S_{b,h}})$ in Algorithm 1. We have

$$U = \sum_{h=1}^{m} [\mathcal{A}(\texttt{rand}; w, \{\xi_i\}_{i \in S_{b,h}})]_C \quad \text{and} \quad U' = \sum_{h=1}^{m} [\mathcal{A}(\texttt{rand}; w, \{\xi'_i\}_{i \in S'_{b,h}})]_C.$$

For data set d, we introduce parameter

$$L(\{S_{b,h}\}_{h=1}^{m}) = |\{1 \leq h \leq m \ : \ \{\xi_i : i \in S_{b,h}\} \cap z \neq \emptyset\}|. \tag{11}$$

Suppose that $L(\{S_{b,h}\}_{h=1}^{m}) = k$. Then exactly $m - k$ terms in the sums of U and U' coincide. Let $h_1, \ldots, h_k$ be the indices of the terms that do not coincide. Then,

$$\begin{aligned} \|U - U'\| &= \| \sum_{h \in \{h_1, \ldots, h_k\}} [\mathcal{A}(\texttt{rand}; w, \{\xi_i\}_{i \in S_{b,h}})]_C - \sum_{h \in \{h_1, \ldots, h_k\}} [\mathcal{A}(\texttt{rand}; w, \{\xi'_i\}_{i \in S'_{b,h}})]_C \| \\ &\leq 2kC. \end{aligned}$$

The norm $\|U - U'\|$ is upper bounded by[¶] $2kC$. This upper bound is met with equality if all the terms happen to be orthogonal to one another. Without further assumptions on how data samples are generated, we must assume this best case for the adversary. That is, we must assume the adversary can possibly perform a hypothesis test of $U + \mathcal{N}(0, (2C\sigma)^2\mathbf{I})$ versus $U' + \mathcal{N}(0, (2C\sigma)^2\mathbf{I})$ where $\|U - U'\| = 2kC$.

After projecting the observed noised local update $\bar{U}$ onto the line that connects U and U' and after normalizing by dividing by $2C$, we have that differentiating whether $\bar{U}$ corresponds to d or d' is equivalent to differentiating whether a received output is from $\mathcal{N}(0, \sigma^2)$ or from $\mathcal{N}(k, \sigma^2)$. Or, equivalently, from $\mathcal{N}(0, 1)$ or from $\mathcal{N}(k/\sigma, 1)$. This is how the Gaussian trade-off function $G_{k/\sigma}$ comes into the picture.

D.5 Round Mechanism with Simulator

Let $\bar{\mathcal{M}}_b$ be the mechanism that produces $\bar{U}$ in round b based on data set d with $L(S_b) = k$. And let $\bar{\mathcal{M}}_b^{\texttt{sim}}$ be the mechanism that represents the mimicked simulator based on data set d'. Then the above argument yields

$$T(\bar{\mathcal{M}}_b(d), \bar{\mathcal{M}}_b^{\texttt{sim}}(d')) = G_{k/\sigma}.$$

D.6 Sampling Distribution

We want to analyse privacy leakage over a whole epoch. For this reason, we are interested in the joint distribution of $\{L(\{S_{b,h}\}_{h=1}^{m})\}_{b=1}^{N/(sm)}$ generated by $\texttt{Sample}_{s,m}$. Let $|z| = g$, that is, d contains g differentiating samples. We define

$$q(c_1, c_2, \ldots, c_g) = \texttt{Pr}[\forall_{k=1}^{g} c_k = \#\{b : L(\{S_{b,h}\}_{h=1}^{m}) = k\} \mid \{S_{b,h}\}_{b=1,h=1}^{N/(sm),m} \leftarrow \texttt{Sample}_{s,m}]. \tag{12}$$

[¶] The original argument for DP-SGD with $s = v = 1$ considers neighboring d and d', that is, $|z \cup z'| = 1$. Update U is computed as the sum $U = \sum_{i \in S_b} [\nabla f(w, \xi_i)]_C$. Their DP argument wrongly states that the vector $x = \sum_{i \in S_b \setminus \{n+1\}} [f(w, \xi_i)]_C$, where $n + 1$ corresponds to the differentiating sample between d and d', is the same for both d and d' and that U for d and d' differs by $[f(w, \xi'_{n+1})]_C$, leading to an upper bound of only C. However, the samples chosen from d and d' corresponding to S_b differ in exactly one element and we need the extra factor 2. This is consistent with our presented analysis. We notice that this was already observed by Dong et al. [2], see p. 25 bottom paragraph. Appendix E has an extended discussion explaining how Opacus [16] implement a slightly different subsampling which is fits the DP-SGD analysis and allows a factor 2 improvement for $g = 1$.

Here, c_k indicates the number of rounds b that have $L(\{S_{b,h}\}_{h=1}^m) = k$ (since the $\{S_{b,h}\}_{h=1}^m$ partition set S_b, $0 \leq k \leq g$).

D.7 Epoch Mechanism with Simulator

Let $(c_1, \ldots, c_g)$ represent an instance of an epoch based on d with $c_k = \#\{b : L(\{S_{b,h}\}_{h=1}^m) = k\}$ for $1 \leq k \leq g$. The probability of such an instance occurring is equal to $q(c_1, c_2, \ldots, c_g)$. Let $\bar{\mathcal{M}}_{c_1,\ldots,c_g}$ be the mechanism that produces a sequence of noised local updates $\bar{U}$ for the given epoch instance based on d. Let $\bar{\mathcal{M}}^{\mathtt{sim}}_{c_1,\ldots,c_g}$ be the mechanism that represents the mimicked simulator based on d'. Then, by composition over rounds within the epoch instance (where we use its commutative property), we conclude that

$$T(\bar{\mathcal{M}}_{c_1,\ldots,c_g}(d), \bar{\mathcal{M}}^{\mathtt{sim}}_{c_1,\ldots,c_g}(d')) = G_{1/\sigma}^{\otimes c_1} \otimes G_{2/\sigma}^{\otimes c_2} \otimes \ldots \otimes G_{g/\sigma}^{\otimes c_g} = G_{\sqrt{\sum_{k=1}^g c_k k^2}/\sigma}. \tag{13}$$

Let $\mathcal{M}$ be the overall mechanism which represents one epoch of updates based on d and let $\mathcal{M}^{\mathtt{sim}}$ represent the mimicked simulator. Then,

$$\mathcal{M} = \sum_{c_1,\ldots,c_g} q(c_1, \ldots, c_g) \cdot \bar{\mathcal{M}}_{c_1,\ldots,c_g} \quad \text{and} \quad \mathcal{M}^{\mathtt{sim}} = \sum_{c_1,\ldots,c_g} q(c_1, \ldots, c_g) \cdot \bar{\mathcal{M}}^{\mathtt{sim}}_{c_1,\ldots,c_g}. \tag{14}$$

The final definition of the mimicked simulator $\mathcal{M}^{\mathtt{sim}}$ has in essence oracle access to the randomness used by $\mathcal{M}$. That is, if $\mathcal{M}$ chooses to use $\bar{\mathcal{M}}_{c_1,\ldots,c_g}$ with probability $q(c_1, \ldots, c_g)$, then this information is passed on to $\mathcal{M}^{\mathtt{sim}}$ who will choose to use $\bar{M}^{\mathtt{sim}}_{c_1,\ldots,c_g}$. It can do this if we assume the strong adversary with oracle access to the randomness used by $\mathcal{M}$ for selecting the sets $S_{b,h}$ and executing $\mathcal{A}$.

D.8 Final Epoch Mechanism

So far, we assumed d to be the truth. If d is the truth, then

$$T(\mathcal{M}(d), \mathcal{M}(d')) \geq T(\mathcal{M}(d), \mathcal{M}^{\mathtt{sim}}(d')). \tag{15}$$

We have $\geq$ if we consider a weaker adversary without direct knowledge about the exact subset S_b and its subsets $S_{b,h}$ with index mappings (for the stronger adversary we have equality). The exact same analysis can be repeated for d' being the truth. In this case we use parameter $g' = |z'|$ and obtain a similar lower bound as given above. We assume the best case for the adversary, that is, the smaller of the two lower bounds.

The larger g the larger z and probability $q(c_1, \ldots, c_g)$ shifts its mass towards vectors $(c_1, \ldots, c_g)$ with larger $c = \sqrt{\sum_{k=1}^g c_k k^2}$, see (11) and (12). This means that the smaller trade-off functions $G_{c/\sigma}$ (corresponding to the larger c) are counted more in mechanism $\mathcal{M}$, see (13) and (14). The best for the adversary is when its hypothesis testing is characterized by the smaller trade-off function (d versus d' being the truth). That is, the one corresponding to the larger of g and g'. Therefore, without loss of generality, we assume $g = |z| \geq |z'|$ and equations (13), (14), and (15) provide a lower bound for $T(\mathcal{M}(d), \mathcal{M}(d'))$. We notice that for our strong adversary we have tight bounds in that (13), (14), and (15) can be met with equality.

Theorem 4.5 in the main body assumes that the effect of $N \neq N'$ leads to privacy leakage small enough to be discarded in our DP analysis as discussed earlier. The theorem statement is for $N \neq N'$ and also uses $g = \max\{|z|, |z'|\}$. In the worst case, the total number $t = |z| + |z'|$ of samples in which d and d' differ are distributed as $|z| = t$ with $|z'| = 0$ as this achieves the maximum $g = |z| + |z'| = t$. This scenario is met if $d = d' + z$ for a group of $g = |z|$ differentiating samples.

D.9 Grouping Probability Mass

We apply Lemma C.1 for (13) and (14) to get for $T(\mathcal{M}(d), \mathcal{M}^{\mathtt{sim}}(d'))(\alpha)$ the lower bound

$$\inf_{\{\alpha_{c_1,\ldots,c_g}\}} \left\{ \sum_{c_1,\ldots,c_g} q(c_1, \ldots, c_g) \cdot G_{\sqrt{\sum_{k=1}^g c_k k^2}/\sigma}(\alpha_{c_1,\ldots,c_g}) \;\middle|\; \sum_{c_1,\ldots,c_g} q(c_1, \ldots, c_g) \cdot \alpha_{c_1,\ldots,c_g} = \alpha \right\}.$$

Now we may apply Lemma C.4 and group probabilities together in the sum of the lower bound that fit the same Gaussian trade-off function:

$$\begin{aligned} q(c) &= \sum_{\{c_k\}_{k=1}^g : c^2 = \sum_{k=1}^g c_k k^2} q(c_1, c_2, \ldots, c_g) \\ &= \texttt{Pr}\left[c^2 = \sum_{k=1}^g \#\{b : L(\{S_{b,h}\}_{h=1}^m) = k\} \cdot k^2 \,\middle|\, \{S_{b,h}\}_{b=1,h=1}^{N/(sm),m} \leftarrow \texttt{Sample}_{s,m} \right]. \end{aligned}$$

This results in

$$T(\mathcal{M}(d), \mathcal{M}^{\texttt{sim}}(d'))(\alpha) \geq f(\alpha) = \inf_{\{\alpha_c\}_{c\in\mathcal{C}}} \left\{ \sum_{c\in\mathcal{C}} q(c) \cdot G_{c/\sigma}(\alpha_c) \,\middle|\, \sum_{c\in\mathcal{C}} q(c) \cdot \alpha_c = \alpha \right\} \tag{16}$$

where

$$\mathcal{C} = \left\{ \sqrt{\sum_{k=1}^g c_k k^2} \;:\; \forall_k \; c_k \in \{0, 1, \ldots, N/(ms)\} \right\}$$

is the set of all possible values c.

Given the previous discussion on g, we notice that f gets smaller for larger g (i.e., the adversary gets better at hypothesis testing).

D.10 Calculation of the Infinum

The infinum can be calculated using Lemma C.5. This yields function f in Theorem 4.6 in the main body.

D.11 Lower and Upper Bounds on $T(\mathcal{M}(d), \mathcal{M}(d'))$ and $T(\mathcal{M}(d), \mathcal{M}^{\texttt{sim}}(d'))$

We apply Lemmas C.2 and C.3 to equations (13) and (14). We use the property that if $c < \hat{c}$, then $G_{\hat{c}/\sigma} < G_{c/\sigma}$.

For the upper bound (the Gaussian trade-off function fits the required assumption in Lemma C.3) we group probabilities together and define

$$u'_{\hat{c}} = \sum_{c\in\mathcal{C} : c < \hat{c}} q(c)$$

Then,

$$u'_{\hat{c}} + (1 - u'_{\hat{c}}) G_{\hat{c}/\sigma}(\min\{1, \alpha/(1 - u'_{\hat{c}})\}) \geq T(\mathcal{M}(d), \mathcal{M}^{\texttt{sim}}(d'))(\alpha).$$

Since Gaussian trade-off functions are decreasing in α and are less than $1 - \alpha$, we have for $u'_{\hat{c}} \leq u_{\hat{c}} \leq 1$

$$u'_{\hat{c}} + (1 - u'_{\hat{c}}) G_{\hat{c}/\sigma}(\min\{1, \alpha/(1 - u'_{\hat{c}})\}) \leq u'_{\hat{c}} + (1 - u'_{\hat{c}}) G_{\hat{c}/\sigma}(\alpha) \leq u_{\hat{c}} + (1 - u_{\hat{c}}) G_{\hat{c}/\sigma}(\alpha).$$

For the lower bound, we define

$$l'_{\hat{c}} = \sum_{c\in\mathcal{C} : c > \hat{c}} q(c)$$

and (by Lemma C.2) we have

$$T(\mathcal{M}(d), \mathcal{M}^{\texttt{sim}}(d'))(\alpha) \geq f(\alpha) \geq G_{\hat{c}/\sigma}(\min\{1, \alpha + l'_{\hat{c}}\}).$$

For $l'_{\hat{c}} \leq l_{\hat{c}} \leq 1$ we have

$$G_{\hat{c}/\sigma}(\min\{1, \alpha + l'_{\hat{c}}\}) \geq G_{\hat{c}/\sigma}(\min\{1, \alpha + l_{\hat{c}}\}).$$

Summarizing,

$$T(\mathcal{M}(d), \mathcal{M}(d')) \geq T(\mathcal{M}(d), \mathcal{M}^{\texttt{sim}}(d')), \tag{17}$$

which is tight for the strong adversary, and we have the lower and upper bounds

$$\hat{f}^{\texttt{U}}_{c^{\texttt{U}}} \geq T(\mathcal{M}(d), \mathcal{M}^{\texttt{sim}}(d')) \geq f \geq \hat{f}^{\texttt{L}}_{c^{\texttt{L}}}, \tag{18}$$

where

$$\begin{aligned} \hat{f}^{\texttt{U}}_{c^{\texttt{U}}}(\alpha) &= u_{c^{\texttt{U}}} + (1 - u_{c^{\texttt{U}}}) G_{c^{\texttt{U}}/\sigma}(\alpha), \\ \hat{f}^{\texttt{L}}_{c^{\texttt{L}}}(\alpha) &= G_{c^{\texttt{L}}/\sigma}(\min\{1, \alpha + l_{c^{\texttt{L}}}\}) \end{aligned}$$

are trade-off functions (continuous, non-increasing, convex, at least 0, at most $1 - \alpha$ for $\alpha \in [0, 1]$).

D.12 f-DP Guarantee

Suppose that $\mathcal{M}$ is h-DP for all pairs of data sets d and d' such that $\max\{|d \setminus d'|, |d' \setminus d|\} \leq g$. Since (17) is tight for the strong adversary, we have

$$T(\mathcal{M}(d), \mathcal{M}(d')) \geq T(\mathcal{M}(d), \mathcal{M}^{\mathtt{sim}}(d')) \geq h. \tag{19}$$

Combined with (18) this implies

$$\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}} \geq T(\mathcal{M}(d), \mathcal{M}^{\mathtt{sim}}(d')) \geq h. \tag{20}$$

By Lemma A.2 in [2],

$$T(\mathcal{M}(d'), \mathcal{M}(d)) = T(\mathcal{M}(d), \mathcal{M}(d'))^{-1}.$$

Since $f \geq h$ implies $f^{-1} \geq h^{-1}$ for non-increasing functions f and h (the inverse preserves order), (19) and (20) prove

$$\hat{f}^{\mathtt{U}\,-1}_{c^{\mathtt{U}}} \geq h^{-1}.$$

We also know that if h is a trade-off function for $\mathcal{M}$, then, by Proposition 2.4 in [2], $\max\{h, h^{-1}\}$ is a trade-off function for $\mathcal{M}$. Therefore, without loss of generality we may assume that h has this form, hence, h is symmetric and we have

$$\hat{f}^{\mathtt{U}\,-1}_{c^{\mathtt{U}}} \geq h^{-1} = h.$$

Together with (20) this gives the upper bound

$$h \leq \min\{\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}, \hat{f}^{\mathtt{U}\,-1}_{c^{\mathtt{U}}}\}.$$

Since h is convex, we can improve the upper bound by means of convexification by using the double conjugate. This yields

$$h \leq \min\{\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}, \hat{f}^{\mathtt{U}\,-1}_{c^{\mathtt{U}}}\}^{**}.$$

We also notice that (18) shows that f is a trade-off function for $\mathcal{M}$. Notice that f is symmetric, see Lemma C.1, hence, from (18) we infer that

$$f = f^{-1} \geq \hat{f}^{\mathtt{L}\,-1}_{c^{\mathtt{L}}}.$$

This proves

$$f \geq \max\{\hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}, \hat{f}^{\mathtt{L}\,-1}_{c^{\mathtt{L}}}\}.$$

D.13 Explicit Formula for the Lower Bound

Since the Gaussian trade-off function is symmetric, $G_{c^{\mathtt{L}}/\sigma}(G_{c^{\mathtt{L}}/\sigma}(x)) = x$ for $x \in [0,1]$. Also notice that the Gaussian trade-off function in x is strictly decreasing and is equal to 1 for $x = 0$ and equal to 0 for $x = 1$. From this we obtain that

$$\hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}(t) = G_{c^{\mathtt{L}}/\sigma}(\min\{1, t + l_{c^{\mathtt{L}}}\}) \leq \alpha$$

if and only if

$$\min\{1, t + l_{c^{\mathtt{L}}}\} \geq G_{c^{\mathtt{L}}/\sigma}(\alpha)$$

if and only if

$$[t \geq 1 - l_{c^{\mathtt{L}}}] \text{ or } [1 - l_{c^{\mathtt{L}}} > t \geq G_{c^{\mathtt{L}}/\sigma}(\alpha) - l_{c^{\mathtt{L}}}].$$

if and only if

$$t \geq G_{c^{\mathtt{L}}/\sigma}(\alpha) - l_{c^{\mathtt{L}}}.$$

This shows that

$$\hat{f}^{\mathtt{L}\,-1}_{c^{\mathtt{L}}}(\alpha) = \inf\{t \in [0,1] \mid \hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}(t) \leq \alpha\} = \max\{0, G_{c^{\mathtt{L}}/\sigma}(\alpha) - l_{c^{\mathtt{L}}}\}.$$

We define $\beta \in [0,1]$ as the solution of

$$G_{c^{\mathtt{L}}/\sigma}(\beta) - l_{c^{\mathtt{L}}} = \beta$$

(there exists a unique solution because $l_{c^{\mathtt{L}}} \leq 1$, $G_{c^{\mathtt{L}}/\sigma}$ is convex, and $G_{c^{\mathtt{L}}/\sigma}(0) = 1$). Notice that

$$\hat{f}^{\mathtt{L}\,-1}_{c^{\mathtt{L}}}(\beta) = \beta = \hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}(\beta)$$

(the second equality follows from $\beta + l_{c^{\mathtt{L}}} = G_{c^{\mathtt{L}}/\sigma}(\beta) \leq 1$, hence, $\hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}(\beta) = G_{c^{\mathtt{L}}/\sigma}(G_{c^{\mathtt{L}}/\sigma}(\beta)) = \beta$).

For $\alpha \in [0, \beta]$,

$$\hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}(\alpha) = G_{c^{\mathtt{L}}/\sigma}(\alpha + l_{c^{\mathtt{L}}}) \leq G_{c^{\mathtt{L}}/\sigma}(\alpha) - l_{c^{\mathtt{L}}},$$

where the inequality follows from the observation that the inequality is met with equality for $\alpha = \beta$ and the derivative of $G_{c^{\mathtt{L}}/\sigma}(\alpha + l_{c^{\mathtt{L}}})$ is at least the derivative of $G_{c^{\mathtt{L}}/\sigma}(\alpha)$ as a function of α due to the convexity of $G_{c^{\mathtt{L}}/\sigma}$. Since $G_{c^{\mathtt{L}}/\sigma}(\alpha) - l_{c^{\mathtt{L}}} \geq G_{c^{\mathtt{L}}/\sigma}(\alpha + l_{c^{\mathtt{L}}}) \geq 0$, we have $\hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}(\alpha) = G_{c^{\mathtt{L}}/\sigma}(\alpha) - l_{c^{\mathtt{L}}}$. We conclude that for $\alpha \in [0, \beta]$,

$$\max\{\hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}, \hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}{}^{-1}\}(\alpha) = G_{c^{\mathtt{L}}/\sigma}(\alpha) - l_{c^{\mathtt{L}}}.$$

By a similar argument, we have that for $\alpha \in [\beta, 1]$,

$$\max\{\hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}, \hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}{}^{-1}\}(\alpha) = G_{c^{\mathtt{L}}/\sigma}(\min\{1, \alpha + l_{c^{\mathtt{L}}}\}).$$

Notice that for $\alpha \in [\beta, 1 - l_{c^{\mathtt{L}}}]$ there exists an $\alpha' \in [0, \beta]$ such that $\alpha = G_{c^{\mathtt{L}}/\sigma}(\alpha') - l_{c^{\mathtt{L}}}$, hence, $G_{c^{\mathtt{L}}/\sigma}(\min\{1, \alpha + l_{c^{\mathtt{L}}}\}) = G_{c^{\mathtt{L}}/\sigma}(G_{c^{\mathtt{L}}/\sigma}(\alpha')) = \alpha'$. This confirms the symmetry of the curve $\max\{\hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}, \hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}{}^{-1}\}(\alpha)$ around the diagonal $\alpha \rightarrow \alpha$.

D.14 Explicit Formula for the Upper Bound

By following the kind of argument we used for improving the lower bound, we observe that

$$\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}(t) = u_{c^{\mathtt{U}}} + (1 - u_{c^{\mathtt{U}}})G_{c^{\mathtt{U}}/\sigma}(t) \leq \alpha$$

if and only if

$$t \geq G_{c^{\mathtt{U}}/\sigma}(\frac{\alpha - u_{c^{\mathtt{U}}}}{1 - u_{c^{\mathtt{U}}}}).$$

This shows that

$$\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}{}^{-1}(\alpha) = \inf\{t \in [0, 1] \mid \hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}(t) \leq \alpha\} = \min\{1, G_{c^{\mathtt{U}}/\sigma}(\frac{\alpha - u_{c^{\mathtt{U}}}}{1 - u_{c^{\mathtt{U}}}})\}.$$

We define $\beta \in [0, 1]$ as the solution of

$$G_{c^{\mathtt{U}}/\sigma}(\frac{\beta - u_{c^{\mathtt{U}}}}{1 - u_{c^{\mathtt{U}}}}) = \beta$$

(there exists a unique solution because $G_{c^{\mathtt{L}}/\sigma}$ is convex, and $G_{c^{\mathtt{L}}/\sigma}(0) = 1$ and $G_{c^{\mathtt{L}}/\sigma}(1) = 0$). Notice that

$$\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}{}^{-1}(\beta) = \beta = \hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}(\beta)$$

(the second equality follows from $G_{c^{\mathtt{L}}/\sigma}(\beta) = G_{c^{\mathtt{L}}/\sigma}(G_{c^{\mathtt{U}}/\sigma}(\frac{\beta - u_{c^{\mathtt{U}}}}{1 - u_{c^{\mathtt{U}}}})) = \frac{\beta - u_{c^{\mathtt{U}}}}{1 - u_{c^{\mathtt{U}}}}$ by the symmetry of $G_{c^{\mathtt{U}}/\sigma}$).

For $\alpha \in [0, \beta]$,

$$\begin{aligned}
\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}(\alpha) &= u_{c^{\mathtt{U}}} + (1 - u_{c^{\mathtt{U}}})G_{c^{\mathtt{U}}/\sigma}(\alpha) \\
&\leq G_{c^{\mathtt{U}}/\sigma}(\frac{\max\{0, \alpha - u_{c^{\mathtt{U}}}\}}{1 - u_{c^{\mathtt{U}}}}) \\
&= \min\{1, G_{c^{\mathtt{U}}/\sigma}(\frac{\alpha - u_{c^{\mathtt{U}}}}{1 - u_{c^{\mathtt{U}}}})\} = \hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}{}^{-1}(\alpha),
\end{aligned}$$

where the inequality follows from the following observation: Since $G_{c^{\mathtt{U}}/\sigma}$ is symmetric, the curve $y = G_{c^{\mathtt{U}}/\sigma}(\frac{x - u_{c^{\mathtt{U}}}}{1 - u_{c^{\mathtt{U}}}})$ for $x \in [u_{c^{\mathtt{U}}}, 1]$ is the same as the curve $x = u_{c^{\mathtt{U}}} + (1 - u_{c^{\mathtt{U}}})G_{c^{\mathtt{U}}/\sigma}(y)$ for $y \in [0, 1]$, hence, it is equal to the curve $y = u_{c^{\mathtt{U}}} + (1 - u_{c^{\mathtt{U}}})G_{c^{\mathtt{U}}/\sigma}(x)$ mirrored around the diagonal $y = x$. The inequality follows from the fact that above the diagonal the mirrored curve is larger than the curve itself (due to the addition of u).

We conclude that for $\alpha \in [0, \beta]$,

$$\min\{\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}, \hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}{}^{-1}\}(\alpha) = u_{c^{\mathtt{U}}} + (1 - u_{c^{\mathtt{U}}})G_{c^{\mathtt{U}}/\sigma}(\alpha)$$

and for $\alpha \in [\beta, 1]$,

$$\min\{\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}, \hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}{}^{-1}\}(\alpha) = G_{c^{\mathtt{U}}/\sigma}(\frac{\alpha - u_{c^{\mathtt{U}}}}{1 - u_{c^{\mathtt{U}}}}).$$

Convexification by means of the double conjugate computes $\beta_0 \leq \beta \leq \beta_1$ such that $\bar{f}^{\mathtt{U}}_{c^{\mathtt{U}}}$ defined as $\min\{\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}, \hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}{}^{-1}\}(\alpha)$ has derivative -1 for $\alpha = \beta_0$ and $\alpha = \beta_1$. Next the upper bound is improved by drawing the line from $(\beta_0, \bar{f}^{\mathtt{U}}_{c^{\mathtt{U}}}(\beta_0))$ to $(\beta_1, \bar{f}^{\mathtt{U}}_{c^{\mathtt{U}}}(\beta_1))$. See (10), $G'_\mu(x) = -e^{\mu\Phi^{-1}(1-x) - \mu^2/2}$. Let $\mu = c^{\mathtt{U}}/\sigma$. This gives

$$\begin{aligned}
-1 &= -(1 - u_{c^{\mathtt{U}}})e^{\mu\Phi^{-1}(1-\beta_0) - \mu^2/2} \\
-1 &= -e^{\mu\Phi^{-1}(1 - \frac{\beta_1 - u_{c^{\mathtt{U}}}}{1 - u_{c^{\mathtt{U}}}}) - \mu^2/2}/(1 - u_{c^{\mathtt{U}}})
\end{aligned}$$

with the solutions

$$\begin{aligned}\beta_0 &= 1-\Phi(\frac{\mu}{2}-\frac{1}{\mu}\ln(1-u_{c^{\mathtt{U}}})),\\ \beta_1 &= u_{c^{\mathtt{U}}}+(1-u_{c^{\mathtt{U}}})\cdot(1-\Phi(\frac{\mu}{2}+\frac{1}{\mu}\ln(1-u_{c^{\mathtt{U}}}))).\end{aligned}$$

Notice that

$$\begin{aligned}\bar{f}^{\mathtt{U}}_{c^{\mathtt{U}}}(\beta_0) &= u_{c^{\mathtt{U}}}+(1-u_{c^{\mathtt{U}}})G_{c^{\mathtt{U}}/\sigma}(\beta_0) = u_{c^{\mathtt{U}}}+(1-u_{c^{\mathtt{U}}})\Phi(-\frac{\mu}{2}-\frac{1}{\mu}\ln(1-u_{c^{\mathtt{U}}})) = \beta_1,\\ \bar{f}^{\mathtt{U}}_{c^{\mathtt{U}}}(\beta_1) &= G_{c^{\mathtt{U}}/\sigma}(\frac{\beta_1-u_{c^{\mathtt{U}}}}{1-u_{c^{\mathtt{U}}}}) = \beta_0.\end{aligned}$$

This proves

$$\min\{\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}},\hat{f}^{\mathtt{U}\,-1}_{c^{\mathtt{U}}}\}^{**}(\alpha)=\begin{cases} u_{c^{\mathtt{U}}}+(1-u_{c^{\mathtt{U}}})G_{c^{\mathtt{U}}/\sigma}(\alpha) & \text{for } \alpha\in[0,\beta_0],\\ \beta_0+\beta_1-\alpha & \text{for } \alpha\in[\beta_0,\beta_1],\\ G_{c^{\mathtt{U}}/\sigma}(\frac{\alpha-u_{c^{\mathtt{U}}}}{1-u_{c^{\mathtt{U}}}}) & \text{for } \alpha\in[\beta_1,1].\end{cases}$$

D.15 Multiple Epochs

For a single epoch, we can use the equivalent formulation

$$L(\{S_{b,h}\}_{h=1}^m)=|\{1\leq h\leq m\ :\ S_{b,h}\cap\{1,\ldots,g\}\neq\emptyset\}|$$

with

$$q(c)=\mathtt{Pr}_{\pi}\left[c^2=\sum_{k=1}^{g}\#\{b:L(\{\pi(S_{b,h})\}_{h=1}^m)=k\}\cdot k^2\ \middle|\ \{S_{b,h}\}_{b=1,h=1}^{N/(sm),m}\leftarrow\mathtt{Sample}_{s,m}\right],$$

where π is a random permutation of $\{1,\ldots,N\}$. This formulation is independent of the actual data set d and makes computing $q(c)$ a combinatorics problem.

All our analysis can be generalized to multiple epochs. Of course, we can use the composition tensor. But an alternative is to take equation (13) and have the c_i do their counting over multiple epochs. This leads to the more general definition

$$q_E(c)=\mathtt{Pr}_{\{\pi^e\}}\left[c^2=\sum_{k=1}^{g}\#\{(b,e):L(\{\pi^e(S^e_{b,h})\}_{h=1}^m)=k\}\cdot k^2\ \middle|\ \{\{S^e_{b,h}\}_{b=1,h=1}^{N/(sm),m}\leftarrow\mathtt{Sample}_{s,m}\}_{e=1}^E\right]$$

and all derivations continue as before. We have summarized our results in Theorems 4.5 and 4.6 in the main body.

E Probabilistic Filtered Batches: Towards Another Factor 2 Improvement

Our algorithmic framework uses static m, s, and v leading to sets S_b with size ms for each round b. We propose a probabilistic filtering of these sets S_b. This will give new sets S'_b with sizes $|S'_b|$ coming from some probability distribution, rather than being constant. As we will see, this may amplify the resulting differential privacy by a factor 2 in a slightly less strong adversarial model.

The main idea is to precompute all $S_{b,h}$ and keep each of the subsets with probability $1/2$. This leads to variable sized subsets

$$S_b=\bigcup_{h\in\mathcal{F}_b}S_{b,h},$$

where $\mathcal{F}_b\subseteq\{1,\ldots,m\}$ with $\mathtt{Pr}[h\in\mathcal{F}_b]=1/2$. Rather than implementing a for loop over $h\in\{1,\ldots,m\}$, we implement a for loop over $h\in\mathcal{F}_b$ in Algorithm 1. We also replace the for loop over $b\in\{1,\ldots,\frac{N}{ms}\}$ by a for loop over

$$b\in\{j\in\{1,\ldots,\frac{N}{ms}\}\mid|\mathcal{F}_b|\geq\tau\}$$

for some threshold τ. For example, when using batch clipping with $m=1$, we only want to keep the rounds with non-empty updates, hence, we set $\tau=1$ (in this case we want to make sure that the stream of updates transmitted to the server correspond to transmission times that follow a memoryless distribution, otherwise the observer can figure out which rounds discard their update).

Notice that the noised update has the form

$$\bar{U}_b = \sum_{h\in\mathcal{F}_b} [a_h]_C + \mathcal{N}(0,(2C\sigma)^2\mathbf{I}) \text{ with } |\mathcal{F}_b| \geq \tau.$$

(We may decide to divide by $|\mathcal{F}_b|$ rather than $\mathbb{E}[|\mathcal{F}_b|] = m/2$ before transmitting to the server.)

In the proof of Theorems 4.5 and 4.6 in Appendix D.4 we compute a sensitivity $2kC$ coming from a bound $\|U_b - U'_b\| \leq 2kC$. Here, the strong adversary replays algorithm $\mathcal{A}$ for the same randomness $\texttt{rand}$ and the same non-differentiating samples in $d \cap d'$ indexed by subsets $S_{b,h}$. Those subsets S_{b_h} that do not include differentiating samples lead to cancellation of terms in the sums that define U_b and U'_b. We only keep the k terms in U_b and the k terms in U'_b based on the $S_{b,h}$ that do contain differentiating samples. This leads to the $2kC$ upper bound.

When we include the proposed filtering $\mathcal{F}_b$, we do not maintain the strong adversarial model which would provide the adversary with the knowledge of all the used coin flips that led to $\mathcal{F}_b$. We only give the adversary knowledge of the subset $\mathcal{S}$ of indices $h \in \mathcal{F}_b$ for which $S_{b,h}$ does not contain any differentiating samples. This means that the adversary only knows set $\mathcal{S}$ with the knowledge that there exists a set $\mathcal{F}_b$ such that $\mathcal{S} \subseteq \mathcal{F}_b$ and $\mathcal{F}_b \setminus \mathcal{S}$ corresponds to indices h for which $S_{b,h}$ contains differentiating sample(s).

The adversary knows that if k subsets $S_{b,h}$ have differentiating samples, then the size $|\mathcal{F}_b \setminus \mathcal{S}| = k'$ with probability $\binom{k}{k'}/2^k$. Only the terms related to the k' indices in $\mathcal{F}_b \setminus \mathcal{S}$ are kept in the sums of U_b and U'_b: We have $\|U_b - U'_b\| \leq 2k'C$ with probability $\binom{k}{k'}/2^k$. In expectation the norm $\|U_b - U'_b\|$ is bounded by $2(k/2)C = kC$.

After adding Gaussian noise, the adversary projects the observed noised local update $\bar{U}_b$ onto the line that connects U_b and U'_b and after normalizing by dividing by $2C$, we have that differentiating whether $\bar{U}_b$ corresponds to d or d' is equivalent to differentiating whether a received output is from $\mathcal{N}(0,\sigma^2)$ or from $\mathcal{N}(k',\sigma^2)$ where $\texttt{Pr}[k'] = \binom{k}{k'}/2^k$ for $0 \leq k' \leq k$. Or equivalently, from $\mathcal{N}(0,1)$ or from $\mathcal{N}(k'/\sigma,1)$ where $\texttt{Pr}[k'] = \binom{k}{k'}/2^k$.

The latter corresponds to a round mechanism $\mathcal{M}$ composed of submechanisms $\mathcal{M}_{k'}$:

$$\mathcal{M} = \sum_{k'=0}^{k} (\binom{k}{k'}/2^k) \cdot \mathcal{M}_{k'},$$

where $\mathcal{M}_{k'}$ is $G_{k'/\sigma}$-DP. We can use the lemmas in Appendix C to analyze bounds on the DP guarantee for $\mathcal{M}$, however, this will not lead to a tight expression.

A better approach is to simply transform distribution $q(c_1,\ldots,c_g)$ as defined in (12). If we have c_k rounds each with k differentiating samples, then the binomial distribution $\texttt{Pr}[k'] = \binom{k}{k'}/2^k$ redistributes these c_k rounds among the c_1, $c_2, \ldots, c_k$. The transformed $q(c_1,\ldots,c_g)$ can be used to compute $q_E(c)$ and the analysis leading to Table 1 can be repeated.

In expectation, the sensitivity is kC and the adversary differentiates between whether a received output is from $\mathcal{N}(0,\sigma^2)$ or from $\mathcal{N}(k/2,\sigma^2)$ (since $\mathbb{E}[k'] = k/2$) and we have the Gaussian trade-off function $G_{k/(2\sigma)}$. This seems to indicate a factor 2 improvement (at best); it is as if σ is increased by a factor 2. We leave a detailed analysis for future work.

For individual clipping with subsampling for $g = 1$, the package Opacus [16] actually implements subsampling using its own filtering technique: For each round b, S_b is populated with samples drawn from a data set of size N where each sample has probability m/N of being included in S_b. This yields a distribution where $|S_b|$ is equal to m in expectation. The adversary does not know whether the differentiating sample is included in S_b or not and this leads to differentiating between U computed on a set S_b and U' computed in a set S_b minus the differentiating sample. This gives the bound $\|U - U'\| \leq C$ rather than $2C$ and we have the factor 2 improvement. This corresponds to DP-SGD [1] which adds $\mathcal{N}(0,(C\sigma)^2\mathbf{I})$ noise without the factor 2 – so the DP analysis in [1] is compatible with the probabilistic filtering implemented in Opacus but misses out on a factor 2 penalty for the original subsampling discussed in [1] and further analysed in this paper. Also, notice that for $g \geq 1$, it is not immediately clear how to prove the factor 2 improvement as we will still want to see a $\sqrt{g}$ dependency in the resulting DP guarantee (although the above intuition seems to also hold for general g).

In the above argument, we *assume* that the adversary does not learn the actually used mini-batch size otherwise we will again need the factor 2 in our analysis. For large m and N, it seems reasonable to assume that the adversary cannot gain significant knowledge about the used mini-batch size from an observed scaled noised update $\bar{U}/m$ (which includes noise $\mathcal{N}(0,(2C\sigma/m)\mathbf{I})$ as a function of the mini-batch size m). Nevertheless, in the Opacus setting the strong adversary in the DP analysis is made slightly weaker.

In this paper we stay on the safe side where we do not attempt a factor 2 improvement by means of probabilistic filtering. This is consistent with the f-DP framework introduced in [2] which avoids discussing a factor 2 improvement.

F Subsampling

In this section we use our main theorems to prove several lemmas which together can be summarized by the next theorem.

Theorem F.1 (Subsampling). *Let h be a trade-off function such that the more general formula (6) with subsampling in Algorithm 1 yields a mechanism $\mathcal{M}$ which is h-DP in the strong adversarial model for all pairs of data sets d and d' with* $\max\{|d \setminus d'|, |d' \setminus d|\} \leq g$ *and $N = |d \cap d'| + g$.*

[Case $g = 1$:] *If we restrict ourselves to $g = 1$, i.e., d and d' are neighboring data sets, then there exists an h such that $\mathcal{M}$ is h-DP and satisfies the lower bound:*

$$G_{\sqrt{(1+1/\sqrt{2E})E}/\sigma}(\min\{1, \alpha + e^{-E}\}) \leq h(\alpha).$$

Furthermore, for all h with the property that $\mathcal{M}$ is h-DP, we have the upper bound

$$h(\alpha) \leq e^{-E} + (1 - e^{-E})G_{\sqrt{(1-1/\sqrt{2E})E}/\sigma}(\alpha).$$

[Case $g = 1$, individual clipping:] *For individual clipping (1) with $g = 1$, mechanism $\mathcal{M}$ is h-DP for*

$$h = C_{m/N}(G_{1/\sigma})^{\otimes (N/m)\cdot E},$$

which is tight [2].

[General g, individual clipping:] *Let*

$$c^{\mathtt{L}} = \sqrt{\beta \min\{m, g\} g E}$$

with $\beta = e^{N/(N-g-m)} + \gamma \approx e + \gamma$ for some $\gamma > 0$, and define

$$l_{c^{\mathtt{L}}} = e^{-\gamma g E}.$$

Then, for individual clipping (1) and general $g \geq 1$, mechanism $\mathcal{M}$ is h-DP for $h = f^{\mathtt{L}}_{c^{\mathtt{L}}}$ with

$$h(\alpha) = f^{\mathtt{L}}_{c^{\mathtt{L}}}(\alpha) \geq \hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}(\alpha) = G_{c^{\mathtt{L}}/\sigma}(\min\{1, \alpha + l_{c^{\mathtt{L}}}\}) = G_{\sqrt{\beta \min\{m,g\} g E}/\sigma}(\min\{1, \alpha + e^{-\gamma g E}\}).$$

[General g, batch clipping:] *Let*

$$q = \frac{N}{s} \cdot (1 - \binom{N-g}{s} / \binom{N}{s}) \approx g$$

for small $s/N \ll 1$. Then, for batch clipping (2) and general $g \geq 1$, mechanism $\mathcal{M}$ is h-DP for some trade-off function h that satisfies the lower bound

$$G_{\sqrt{(1+1/\sqrt{2qE})qE}/\sigma}(\min\{1, \alpha + e^{-qE}\}) \leq h(\alpha).$$

Furthermore, for all h with the property that $\mathcal{M}$ is h-DP, we have the upper bound

$$h(\alpha) \leq e^{-qE} + (1 - e^{-qE})G_{\sqrt{(1-1/\sqrt{2qE})qE}/\sigma}(\alpha).$$

F.1 Complexity of $q_E(c)$

We first consider individual clipping with subsampling, that is, $s = v = 1$. Since $s = v = 1$, each set $S^e_{b,h}$ is a singleton set and the union $S^e_b = \bigcup_{h=1}^m S^e_{b,h}$ is a set of size m. We have

$$L(\{\pi^e(S^e_{b,h})\}_{h=1}^m) = |\pi^e(S^e_b) \cap \{1, \ldots, g\}|.$$

Due to the random permutation π^e, $\pi^e(S^e_b)$ is randomly subsampled from the set of indices $\{1, \ldots, N\}$. Therefore, for $k \leq m$,

$$\mathtt{Pr}[L(\{\pi^e(S^e_{b,h})\}_{h=1}^m) = k] = \binom{N-g}{m-k}\binom{g}{k} / \binom{N}{m}.$$

We denote this probability by q_k. For $m < k \leq g$, it is straightforward to see that $\mathtt{Pr}[L(\{\pi^e(S^e_{b,h})\}_{h=1}^m) = k] = 0$ and we define $q_k = 0$ for $m < k \leq g$.

Given this notation, the joint probability of having

$$\{c_k = \#\{(b, e) : L(\{\pi^e(S^e_{b,h})\}_{h=1}^m) = k\}\}_{k=1}^g$$

is equal to

$$q_E(c_1,\ldots,c_g) = \binom{(N/m)\cdot E}{c_1,c_2,\ldots,c_g}(1-\sum_{k=1}^{g} q_k)^{(N/m)\cdot E-\sum_{k=1}^{g} c_k}\prod_{k=1}^{g} q_k^{c_k}. \tag{21}$$

We can use these probabilities in the definition of trade-off function f, but this becomes too complex. Even the upper and lower bounds will require some fine tuned calculus in order to get tight expressions.

For the general setting of formula (6) we need to consider $s > 1$. This means that the probabilities q_k become more complex. Now ms samples are selected out of the data set with N samples and grouped in m subsets $S^e_{b,h}$ of size s. The total number of possibilities is

$$\binom{N}{ms}\binom{ms}{s,s,\ldots,s}.$$

Suppose that k' of the ms samples belong to the g differentiating samples – there are

$$\binom{N-g}{ms-k'}\binom{g}{k'}$$

combinations. Suppose that k'_h differentiating samples are put in the subset $S^e_{b,h}$ of size s. Then, $k' = \sum_{h=1}^{m} k'_h$ with $0 \leq k'_h \leq s$. This gives

$$\binom{k'}{k'_1,\ldots,k'_m}$$

combinations. Notice that $s - h'_h$ non-differentiating samples are put in $S^e_{b,h}$. This gives

$$\binom{ms-k'}{(s-k'_1),\ldots,(s-k'_m)}$$

combinations. If $k = \#\{h \mid k_h \neq 0\}$, then we contribute to q_k. Therefore, let

$$\mathcal{K}_k = \{(k'_1,\ldots,k'_m) \mid k = \#\{h \mid k_h \neq 0\} \text{ and } \forall_{h=1}^{m} 0 \leq k'_h \leq s\}.$$

Combining all previous expressions yields

$$q_k = \frac{\sum_{(k'_1,\ldots,k'_m)\in\mathcal{K}_k}\binom{N-g}{ms-\sum_{h=1}^{m} k'_h}\binom{g}{\sum_{h=1}^{m} k'_h}\binom{\sum_{h=1}^{m} k'_h}{k'_1,\ldots,k'_m}\binom{ms-\sum_{h=1}^{m} k'_h}{(s-k'_1),\ldots,(s-k'_m)}}{\binom{N}{ms}\binom{ms}{s,s,\ldots,s}},$$

a considerably more complex expression for q_k to work with.

F.2 Individual Clipping for $g = 1$

We start by analyzing individual clipping for the simplest case $g = 1$. We will show that we can derive simple to interpret tight formulas that resemble the Gaussian trade-off function $G_{\sqrt{E}/\sigma}$. In the case $g = 1$ we only need to consider $q_1 = m/N$. In $q_E(c_1,\ldots,c_g)$ notation this gives

$$q_E(c_1) = \binom{(N/m)\cdot E}{c_1}(1-m/N)^{(N/m)\cdot E-c_1}(m/N)^{c_1}.$$

Notice that the theorem uses $q_E(c)$ which equals the above $q_E(c_1)$ evaluated in $c_1 = c^2$, that is, $c = \sqrt{c_1}$. From Hoeffding's inequality we obtain

$$\begin{aligned}\sum_{c'_1 \leq c_1} q_E(c'_1) &\leq \exp(-2(E-c_1)^2) \text{ for } c_1 \leq E,\\ \sum_{c'_1 \geq c_1} q_E(c'_1) &\leq \exp(-2(E-c_1)^2) \text{ for } c_1 \geq E.\end{aligned}$$

In the upper and lower bounds of Theorem 4.5 we choose $(c^{\mathtt{U}})^2 = (1-1/\sqrt{2E})E$ which gives $u_{c^{\mathtt{U}}} = e^{-E}$ as upper bound on the first sum and we choose $(c^{\mathtt{L}})^2 = (1+1/\sqrt{2E})E$ which gives $l_{c^{\mathtt{L}}} = e^{-E}$ as upper bound on the second sum.

From this we conclude that

$$\begin{aligned} \hat{f}^{\mathtt{U}}_{C^{\mathtt{U}}}(\alpha) &= e^{-E} + (1-e^{-E}) G_{\sqrt{(1-1/\sqrt{2E})E}/\sigma}(\alpha), \\ \hat{f}^{\mathtt{L}}_{C^{\mathtt{L}}}(\alpha) &= G_{\sqrt{(1+1/\sqrt{2E})E}/\sigma}(\min\{1, \alpha + e^{-E}\}). \end{aligned}$$

For larger E, the upper and lower bound converge to $G_{\sqrt{E}/\sigma}(\alpha)$. We recall that for the strong adversary $h = C_{m/N}(G_{1/\sigma})^{\otimes (N/m)\cdot E}$ is a tight trade-off function for $\mathcal{M}$ with $g = 1$.

We notice that the above analysis for $g = 1$ also holds for the mechanism based on the general formula (6) (by replacing m with sm).

Lemma F.2. *Individual clipping (1) with subsampling in Algorithm 1 (i.e., DP-SGD) yields a mechanism $\mathcal{M}$ which is $C_{m/N}(G_{1/\sigma})^{\otimes (N/m)\cdot E}$-DP for all pairs of data sets d and d′ with* $\max\{|d \setminus d'|, |d' \setminus d|\} \leq 1$ *and* $N = |d \cap d'| + 1$ *(we have $g = 1$). This is a tight trade-off function in the strong adversarial model with*

$$\begin{aligned} & G_{\sqrt{(1+1/\sqrt{2E})E}/\sigma}(\min\{1, \alpha + e^{-E}\}) \\ \leq\ & C_{m/N}(G_{1/\sigma})^{\otimes (N/m)\cdot E}(\alpha) \\ \leq\ & e^{-E} + (1-e^{-E}) G_{\sqrt{(1-1/\sqrt{2E})E}/\sigma}(\alpha). \end{aligned}$$

The lemma provides a *simple* formula for the analysis of the original DP-SGD in terms of the Gaussian trade-off function G_μ. The composition tensor and subsampling operator do not play a role in the lower and upper bounds. This is big step forward to being able to immediately interpret the consequences of setting system parameters for the DP guarantee. A differential privacy accountant for computing the effect of composition is not needed.

The lemma can be improved by using the lower and upper bounds derived from the symmetric trade-off functions $f^{\mathtt{L}}_{C^{\mathtt{L}}}$ and $f^{\mathtt{U}}_{C^{\mathtt{U}}}$. However, the weaker bounds in the current lemma already capture intuitions stated in the next corollaries. The first intuition gained from the lemma is that, for large E, $C_{m/N}(G_{1/\sigma})^{\otimes (N/m)\cdot E}$ converges to $G_{\sqrt{E}/\sigma}$. In precise mathematical terms we have the following corollary.

Corollary F.3. *For $E \to \infty$ with $\sqrt{E}/\sigma \to \mu$ for some constant $\mu > 0$, we have*

$$C_{m/N}(G_{1/\sigma})^{\otimes (N/m)\cdot E} \to G_\mu.$$

Also, we notice that the lower and upper bounds are independent from N. Therefore, the upper and lower bounds in the lemma hold for *all* neighboring data sets regardless their size.

Corollary F.4. *$\mathcal{M}$ is $G_{\sqrt{(1+1/\sqrt{2E})E}/\sigma}(\min\{1, \alpha + e^{-E}\})$-DP for all data sets d and d′ for which* $\max\{|d \setminus d'|, |d' \setminus d|\} \leq 1$.

Another main insight is that the lower and upper bounds do not depend on the actual number of rounds performed by DP-SGD within E epochs. This strengthens the last corollary.

Corollary F.5. *The DP guarantee of Corollary F.4 holds for all m, that is, assuming that an epoch computes N gradients over N samples, $\mathcal{M}$ can freely choose the number of rounds N/m in each epoch of length N. This can be used to optimize for best accuracy.*

The latter property comes from the fact that in expectation a single epoch will only leak privacy related to one $G_{1/\sigma}$ instance. This is because there are N/m rounds and each round has probability m/N to leak privacy according to $G_{1/\sigma}$ (subsampling with probability m/N and composition over N/m rounds cancels one another). Composition over E epochs will lead to $G_{1/\sigma}^{\otimes E} = G_{\sqrt{E}/\sigma}$ in expectation. The resulting independence of the total number of rounds gives the engineer the freedom to tune parameter m for achieving the best accuracy.

F.3 General Clipping for $g = 1$

We notice that the analysis leading to Lemma F.2 for $g = 1$ also holds for the mechanism based on the general formula (6) (by replacing m with sm):

Lemma F.6. *The more general formula (6) with subsampling in Algorithm 1 yields a mechanism $\mathcal{M}$ which is h-DP in the strong adversarial model for all pairs of data sets d and d′ with* $\max\{|d \setminus d'|, |d' \setminus d|\} \leq 1$ *and* $N = |d \cap d'| + 1$ *(we have $g = 1$) for some trade-off function h satisfying the lower bound:*

$$G_{\sqrt{(1+1/\sqrt{2E})E}/\sigma}(\min\{1, \alpha + e^{-E}\}) \leq h(\alpha).$$

Furthermore, for all $\bar{h}$ with the property that $\mathcal{M}$ is $\bar{h}$-DP, we have the upper bound

$$\bar{h}(\alpha) \leq e^{-E} + (1 - e^{-E}) G_{\sqrt{(1-1/\sqrt{2E})E}/\sigma}(\alpha).$$

This shows into what extent the lower bound is tight.

This lemma leads to similar Corollaries F.3, F.4, and F.5 as the ones discussed above.

F.4 Lower Bound for Individual Clipping for g

In order to get a lower bound on the trade-off function for individual clipping ($s = v = 1$) with subsampling that holds for arbitrary g, we derive and use upper bounds on the q_k (this puts more weight towards larger k, hence, larger c in $q_E(c)$, which in turn favors the smaller Gaussian trade-off functions $G_{c/\sigma}$).

We first consider $g > m$. Notice that $q_k = 0$ for $m < k \leq g$. In the worst-case we see scaling with m in a single round leading to trade-off function $G_{m/\sigma}$ (see the group privacy discussion in Section A). In expectation there are at most g rounds that have differentiating samples. Therefore, a coarse worst-case analysis should lead to a lower bound of $G_{m/\sigma}^{\otimes g}$ for the trade-off function for a single epoch. composition over E epochs gives the lower bound trade-off function $G_{m\sqrt{g}\sqrt{E}/\sigma}$. As we will see, a precise analysis can improve this to a $\sqrt{mg}$ rather than a $m\sqrt{g}$ dependency:

For individual clipping we have

$$\begin{aligned}
q_{k+1} &= \binom{N-g}{m-k-1}\binom{g}{k+1} / \binom{N}{m} \\
&= \binom{N-g}{m-k}\frac{m-k}{N-g-m+k+1}\binom{g}{k}\frac{g-k}{k+1} / \binom{N}{m} \\
&= q_k \frac{m-k}{N-g-m+k+1}\frac{g-k}{k+1} \\
&\leq q_k \frac{g}{N-g-m}\frac{m-k}{k+1}.
\end{aligned}$$

and

$$\begin{aligned}
q_1 &= \binom{N-g}{m-1}\binom{g}{1} / \binom{N}{m} \\
&= \frac{gm}{N}\binom{N-g}{m-1} / \binom{N-1}{m-1} \leq \frac{gm}{N} \leq \frac{g}{N-g-m}\frac{m}{1}.
\end{aligned}$$

By induction in k,

$$q_k \leq (\frac{g}{N-g-m})^k \binom{m}{k}.$$

We substitute this in (21) and obtain

$$\begin{aligned}
q_E(c_1, \ldots, c_m, 0, ..., 0) &\leq \binom{(N/m) \cdot E}{c_1, c_2, \ldots, c_m} \prod_{k=1}^{m} q_k^{c_k} \\
&\leq \binom{(N/m) \cdot E}{c_1, c_2, \ldots, c_m} (\frac{g}{N-g-m})^{\sum_{k=1}^m c_k \cdot k} \prod_{k=1}^{m} \binom{m}{k}^{c_k}.
\end{aligned}$$

Let

$$c^{\mathrm{L}} = \sqrt{\beta m g E} \text{ with } \beta = e^{N/(N-g-m)} + \gamma \approx e + \gamma,$$

for some $\gamma > 0$. For $c = \sqrt{\sum_{k=1}^m c_k \cdot k^2} > c^{\mathrm{L}}$, we have

$$\sum_{k=1}^{m} c_k \cdot k \geq \sum_{k=1}^{m} c_k \cdot k^2/m \geq \beta m g E/m = \beta g E.$$

Therefore, if $c > c^{\mathrm{L}}$, then

$$\begin{aligned}
(\frac{g}{N-g-m})^{\sum_{k=1}^m c_k \cdot k} &= (\frac{g\beta}{(N-g-m)})^{\sum_{k=1}^m c_k \cdot k}(1/\beta)^{\sum_{k=1}^m c_k \cdot k} \\
&\leq (\frac{g\beta}{(N-g-m)})^{\sum_{k=1}^m c_k \cdot k}(1/\beta)^{\beta g E}.
\end{aligned}$$

This proves that

$$\begin{aligned}
& \sum_{c_1,c_2,\dots c_m : c=\sqrt{\sum_{k=1}^m c_k\cdot k^2}>c^{\mathtt{L}}} q_E(c_1,\dots,c_m,0,...,0) \\
\leq \quad & \sum_{c_1,c_2,\dots,c_m} \binom{(N/m)\cdot E}{c_1,c_2,\dots,c_m} (\frac{g\beta}{(N-g-m)})^{\sum_{k=1}^m c_k\cdot k} (1/\beta)^{\beta gE} \prod_{k=1}^m \binom{m}{k}^{c_k} \\
= \quad & (1/\beta)^{\beta gE} \sum_{c_1,c_2,\dots,c_m} \binom{(N/m)\cdot E}{c_1,c_2,\dots,c_m} \prod_{k=1}^m (\binom{m}{k}(\frac{g\beta}{(N-g-m)})^k)^{c_k} \\
= \quad & (1/\beta)^{\beta gE}(1+\sum_{k=1}^m \binom{m}{k}(\frac{g\beta}{(N-g-m)})^k)^{(N/m)\cdot E} \\
= \quad & (1/\beta)^{\beta gE}((1+\frac{g\beta}{(N-g-m)})^m)^{(N/m)\cdot E} \\
= \quad & (1/\beta)^{\beta gE}(1+\frac{g\beta}{(N-g-m)})^{NE} \\
\leq \quad & (1/\beta)^{\beta gE} e^{(g\beta)(N/(N-g-m))E} \\
= \quad & ((e^{(N/(N-g-m)}/\beta)^{\beta gE} = (1-\gamma/\beta)^{\beta gE} \leq e^{-\gamma gE},
\end{aligned}$$

where the two last inequalities follow from $(1+x/y)^y \leq e^x$ for all real valued $x > -y$ (by definition $-\gamma > -\beta$).

Notice that the above analysis can be generalized to $g \leq m$: We substitute m by g in the definition of probability $q_E(\dots)$,

$$q_E(c_1,\dots,c_g) \leq \binom{(N/m)\cdot E}{c_1,c_2,\dots,c_g} \prod_{k=1}^g q_k^{c_k},$$

and we use $c^{\mathtt{L}} = \sqrt{\beta g^2 E}$ such that we can repeat the argument $\sum_{k=1}^g c_k \cdot k \geq \sum_{k=1}^g c_k \cdot k^2/g \geq \beta g^2 E/g = \beta gE$ as before. We obtain a linear g dependency, which is also what the group privacy analysis with operator $\circ$ would give us. It remains an open problem to improve the analysis for $g \leq m$.

Application of Theorem 4.6 gives the following lemma:

Lemma F.7. *Let*

$$c^{\mathtt{L}} = \sqrt{\beta \min\{m,g\} gE}$$

with $\beta = e^{N/(N-g-m)} + \gamma \approx e + \gamma$ *for some* $\gamma > 0$*, and define*

$$l_{c^{\mathtt{L}}} = e^{-\gamma gE}.$$

Then, individual clipping (1) with subsampling in Algorithm 1 (i.e., DP-SGD) yields a mechanism $\mathcal{M}$ *which is* h*-DP for* $h = f^{\mathtt{L}}_{c^{\mathtt{L}}}$ *with*

$$h(\alpha) = f^{\mathtt{L}}_{c^{\mathtt{L}}}(\alpha) \geq \hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}}(\alpha) = G_{c^{\mathtt{L}}/\sigma}(\min\{1, \alpha + l_{c^{\mathtt{L}}}\}) = G_{\sqrt{\beta \min\{m,g\} gE}/\sigma}(\min\{1, \alpha + e^{-\gamma gE}\})$$

in the strong adversarial model for all pairs of data sets d *and* d' *with* $\max\{|d \setminus d'|, |d' \setminus d|\} \leq g$ *and* $N = |d \cap d'| + g$*.*

We again have analogues of Corollaries F.4 and F.5. And we stress the $\sqrt{g}$ dependency in the following corollary:

Lemma F.2 for $g = 1$ shows that $\mathcal{M}$ is $G_{\sqrt{(1+1/\sqrt{2E})E}/\sigma}(\min\{1, \alpha + e^{-E}\})$-DP. For $E \gg 1$, this is approximately equivalent to $G_{\sqrt{(1+1/\sqrt{2E})E}/\sigma}$-DP. The group privacy analysis based on operator $\circ$ shows that, for general $g \geq 1$, $\mathcal{M}$ is $G_{g\sqrt{(1+1/\sqrt{2E})E}/\sigma}$-DP.

Lemma F.7 on the other hand shows that $\mathcal{M}$ is $G_{\sqrt{\beta \min\{m,g\} gE}/\sigma}(\min\{1, \alpha + e^{-\gamma gE}\})$-DP for constants $\beta \approx e + \gamma$ with $\gamma > 0$. Let $\gamma = 1/g$. Then $\mathcal{M}$ has trade-off function

$$G_{\sqrt{\beta \min\{m,g\} gE}/\sigma}(\min\{1, \alpha + e^{-\gamma gE}\}) \approx G_{\sqrt{(e+1/g) \min\{m,g\} gE}/\sigma}(\alpha).$$

Corollary F.8. *For* $\gamma = 1/g$ *and* $E \gg 1$*,* $\mathcal{M}$ *in Lemma F.7 is approximately* G_μ*-DP with*

$$\mu = \sqrt{(e+1/g)\min\{m,g\} \cdot gE}/\sigma.$$

The group privacy analysis based on the $\circ$ operator and Lemma F.2 yields $G_{\mu'}$-DP with

$$\mu' = \sqrt{(1+1/\sqrt{2E})g \cdot gE}/\sigma.$$

For $g > m$, μ scales with $\sqrt{g}$ while μ' scales with g implying that lower bound G_μ will outperform $G_{\mu'}$ (since it will be closer to $G_0(\alpha) = 1 - \alpha$).

F.5 Batch Clipping for g

We consider batch clipping with $m = 1$ and $v = 1$. This gives N/s rounds within an epoch and each round computes on a batch of s data samples. We have

$$L(\{\pi^e(S^e_{b,h})\}_{h=1}^1) = \begin{cases} 1, & \text{if } |\pi^e(S^e_b) \cap \{1,\ldots,g\}| \neq 0, \\ 0, & \text{if } |\pi^e(S^e_b) \cap \{1,\ldots,g\}| = 0. \end{cases}$$

We have

$$\begin{aligned} \mathtt{Pr}[L(\{\pi^e(S^e_{b,h})\}_{h=1}^1) = 1] &= 1 - \binom{N-g}{s} / \binom{N}{s} = 1 - \prod_{j=0}^{g-1} \frac{N-s-j}{N-j} \\ &\approx 1 - (1 - s/N)^g \approx gs/N. \end{aligned}$$

We denote this probability by q_1 and define $q_2 = \ldots = q_g = 0$. Given this notation, the probability of having

$$c_1 = \#\{(b,e) : L(\{\pi^e(S^e_{b,h})\}_{h=1}^1) = 1\}$$

is equal to

$$q_E(c_1, c_2 = 0, \ldots, c_g = 0) = \binom{(N/s)\cdot E}{c_1} (1-q_1)^{(N/s)\cdot E - c_1} q_1^{c_1}.$$

We repeat the same type of calculus as before. From Hoeffding's inequality we obtain

$$\begin{aligned} \sum_{c_1' \le c_1} q_E(c_1', 0, \ldots, 0) &\le \exp(-2((N/s)Eq_1 - c_1)^2) \text{ for } c_1 \le (N/s)Eq_1, \\ \sum_{c_1' \ge c_1} q_E(c_1', 0, \ldots, 0) &\le \exp(-2((N/s)Eq_1 - c_1)^2) \text{ for } c_1 \ge (N/s)Eq_1. \end{aligned}$$

In the upper and lower bounds of Theorem 4.5 we choose $(c^{\mathtt{U}})^2 = (1 - 1/\sqrt{2(N/s)Eq_1})(N/s)Eq_1$ which gives $u_{c^{\mathtt{U}}} = e^{-(N/s)Eq_1}$ and we choose $(c^{\mathtt{L}})^2 = (1 + 1/\sqrt{2(N/s)Eq_1})(N/s)Eq_1$ which gives $l_{c^{\mathtt{L}}} = e^{-(N/s)Eq_1}$. Notice that $q_1 \approx gs/N$ which makes $(N/s)Eq_1 \approx gE$.

Lemma F.9. *Let*

$$q = \frac{N}{s} \cdot (1 - \binom{N-g}{s} / \binom{N}{s}) \approx g$$

for small $s/N \ll 1$. Then, batch clipping with subsampling in Algorithm 1 yields a mechanism $\mathcal{M}$ which is h-DP in the strong adversarial model for all pairs of data sets d and d' with $\max\{|d \setminus d'|, |d' \setminus d|\} \le g$ and $N = |d \cap d'| + g$ for some trade-off function h satisfying the lower bound:

$$G_{\sqrt{(1+1/\sqrt{2qE})qE}/\sigma}(\min\{1, \alpha + e^{-qE}\}) \le h(\alpha).$$

Furthermore, for all $\bar{h}$ with the property that $\mathcal{M}$ is $\bar{h}$-DP, we have the upper bound

$$\bar{h}(\alpha) \le e^{-qE} + (1 - e^{-qE}) G_{\sqrt{(1-1/\sqrt{2qE})qE}/\sigma}(\alpha).$$

This shows into what extent the lower bound is tight.

We have analogues of Corollaries F.3, F.4, F.5, and F.8 on the behavior of the DP guarantee which tends to $G_{\sqrt{(1-1/\sqrt{2qE})qE}/\sigma}$-DP where $q \approx g$, on the approximate independence on N through $q \approx g$, on the independence of the number of rounds per epoch, and the $\sqrt{q} \approx \sqrt{g}$ dependency for group privacy.

Notice that for $1 < g < m$ batch clipping yields a better DP guarantee compared to the DP analysis of individual clipping in Lemma F.7 and Corollary F.8.

G Shuffling

In this section we use our main theorems to prove several lemmas which together can be summarized by the next theorem.

Theorem G.1 (**Shuffling).** *Let h be a trade-off function such that the more general formula (6) with shuffling in Algorithm 1 yields a mechanism $\mathcal{M}$ which is h-DP in the strong adversarial model for all pairs of data sets d and d' with $\max\{|d \setminus d'|, |d' \setminus d|\} \leq g$ and $N = |d \cap d'| + g$.*

*[**Case** $g = 1$**:**] If we restrict ourselves to $g = 1$, i.e., d and d' are neighboring data sets, then $\mathcal{M}$ is h-DP for*

$$h = G_{\sqrt{E}/\sigma}$$

(lower bound). Furthermore, this is tight and cannot be improved (upper bound).

*[**Case** $g \geq 1$**:**] Let $c^{\mathtt{L}} = \sqrt{g}$ and $l_{c^{\mathtt{L}}} = g^2 ms/(N - g)$. Then, mechanism $\mathcal{M}$ is $h^{\otimes E}$-DP for $h = f^{\mathtt{L}}_{c^{\mathtt{L}}}$ with*

$$h = f^{\mathtt{L}}_{c^{\mathtt{L}}} \geq \hat{f}^{\mathtt{L}}_{c^{\mathtt{L}}} = G_{c^{\mathtt{L}}/\sigma}(\min\{1, \alpha + l_{c^{\mathtt{L}}}\}) = G_{\sqrt{g}/\sigma}(\min\{1, \alpha + \frac{g^2 ms}{N - g}\})$$

and

$$h^{\otimes E} = {f^{\mathtt{L}}_{c^{\mathtt{L}}}}^{\otimes E} \geq (\alpha \to G_{\sqrt{g}/\sigma}(\min\{1, \alpha + \frac{g^2 ms}{N - g}\}))^{\otimes E} \approx G_{\sqrt{gE}/\sigma},$$

where the approximation is for constant E and small $g^2 ms/(N - g)$.

*[**General** g**, batch clipping:**] For batch clipping (2) and general $g \geq 1$, mechanism $\mathcal{M}$ is h-DP for*

$$h = G_{\sqrt{gE}/\sigma}$$

(lower bound). For $g \leq s$, mechanism $\mathcal{M}$ is h-DP for $h = f^{\otimes E}$ with

$$h = f^{\otimes E} \geq G_{\sqrt{gE}/\sigma},$$

f is defined as the symmetric trade-off function

$$f(\alpha) = \sum_{j=1}^{g} q_j \cdot \Phi(\Lambda(\alpha) \cdot \frac{\sigma}{\sqrt{j}} - \frac{\sqrt{j}}{2\sigma}) \text{ with } 1 - \alpha = \sum_{j=1}^{g} q_j \cdot \Phi(\Lambda(\alpha) \cdot \frac{\sigma}{\sqrt{j}} + \frac{\sqrt{j}}{2\sigma}),$$

where $q_j = \binom{N/s}{j}\binom{g-1}{j-1}/\binom{N/s-1+g}{g}$.

Furthermore, let $c^{\mathtt{U}} = \sqrt{g}$ and $u_{c^{\mathtt{U}}} = g^2/(N/s - g - g^2)$. If $g \leq s$, then for all h with the property that $\mathcal{M}$ is h-DP, we have the upper bound

$$h \leq {f^{\mathtt{U}}_{c^{\mathtt{U}}}}^{\otimes E} \leq {\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}}^{\otimes E},$$

where

$${\hat{f}^{\mathtt{U}}_{c^{\mathtt{U}}}}^{\otimes E} = (u_{c^{\mathtt{U}}} + (1 - u_{c^{\mathtt{U}}}) G_{c^{\mathtt{U}}/\sigma})^{\otimes E} \approx G^{\otimes E}_{\sqrt{g}/\sigma} = G_{\sqrt{gE}/\sigma}$$

for constant E and small $g^2/(N/s - g - g^2)$.

G.1 General Clipping for $g = 1$

Consider a single epoch. For $g = 1$, we have that exactly one round in the epoch has the differentiating sample. We do not even need to use our main theorems and can immediately conclude that the trade-off function is equal to $G_{1/\sigma}$ for each epoch. This is tight for the strong adversary. Composing this over E rounds gives $G_{\sqrt{E}/\sigma}$ and fits Lemma F.6.

Lemma G.2. *The more general formula (6) with shuffling in Algorithm 1 yields a mechanism $\mathcal{M}$ which is $G_{\sqrt{E}/\sigma}$-DP for all pairs of data sets d and d' with $\max\{|d \setminus d'|, |d' \setminus d|\} \leq 1$ and $N = |d \cap d'| + 1$ (we have $g = 1$). This is tight in the strong adversarial model.*

G.2 General clipping for $g \geq 1$

We will now consider the general case $g \geq 1$ for a single epoch $e = 1$ and prove a lower bound based on Theorem 4.6. Let $c^{\mathrm{L}} = \sqrt{g}$. Notice that if each of the g differentiating samples are separated by at least $ms - 1$ non-differentiating samples in the permutation that defines the shuffling of all data samples used for the epoch, then each subset $\pi^e(S_b)$ has at most one differentiating sample.[‖] This means that there are exactly g rounds that each have one differentiating sample, hence, $c_1 = g$ and $c_2 = \ldots = c_g = 0$.

The probability that each of the g differentiating samples are separated by at least $ms - 1$ non-differentiating samples is at least the number of combinations for distributing $N - gms$ non-differentiating samples and g groups of one differentiating sample followed by $ms - 1$ non-differentiating samples, divided by the total number of ways g differentiating samples can be distributed among N samples: We have

$$\begin{aligned}
q_{E=1}(c_1 = g, 0, \ldots, 0) &\geq \binom{N - gms + g}{g} / \binom{N}{g} \\
&= \frac{N - gms + g}{N} \cdot \ldots \cdot \frac{N - gms + 1}{N - g + 1} \\
&\geq (\frac{N - gms + 1}{N - g + 1})^g = (1 - g(ms - 1)/(N - g + 1))^g \\
&\geq 1 - g^2(ms - 1)/(N - g + 1) \geq 1 - g^2 ms/(N - g).
\end{aligned}$$

We derive

$$\sum_{c > c^{\mathrm{L}}} q_E(c) = 1 - \sum_{c \leq c^{\mathrm{L}} = \sqrt{ag}} q_E(c) \leq 1 - q_{E=1}(c_1 = g, 0, \ldots, 0) \leq g^2 ms/(N - g).$$

This shows that we can apply Theorem 4.6 for $c^{\mathrm{L}} = \sqrt{g}$ and $l_{c^{\mathrm{L}}} = g^2 ms/(N - g)$. We obtain the following lemma, where we use composition over E epochs.

Lemma G.3. *Let*

$$c^{\mathrm{L}} = \sqrt{g} \textit{ and } l_{c^{\mathrm{L}}} = g^2 ms/(N - g).$$

Then, the more general formula (6) with shuffling in Algorithm 1 for one epoch $E = 1$ yields a mechanism $\mathcal{M}$ which is h-DP for $h = f^{\mathrm{L}}_{c^{\mathrm{L}}}$ with

$$h = f^{\mathrm{L}}_{c^{\mathrm{L}}} \geq \hat{f}^{\mathrm{L}}_{c^{\mathrm{L}}} = G_{c^{\mathrm{L}}/\sigma}(\min\{1, \alpha + l_{c^{\mathrm{L}}}\}) = G_{\sqrt{g}/\sigma}(\min\{1, \alpha + \frac{g^2 ms}{N - g}\})$$

in the strong adversarial model for all pairs of data sets d and d' with $\max\{|d \setminus d'|, |d' \setminus d|\} \leq g$ and $N = |d \cap d'| + g$. By using composition over E epochs this yields the lower bound

$$h^{\otimes E} = {f^{\mathrm{L}}_{c^{\mathrm{L}}}}^{\otimes E} \geq (\alpha \to G_{\sqrt{g}/\sigma}(\min\{1, \alpha + \frac{g^2 ms}{N - g}\}))^{\otimes E} \approx G_{\sqrt{gE}/\sigma},$$

where the approximation is for constant E and small $g^2 ms/(N - g)$.

G.3 Batch Clipping for g

We consider a single epoch, $E = 1$. Since $m = 1$, we only need to consider values c_1 (values $c_2 = \ldots = c_g = 0$). We assume $g \leq s$. Then, the total number of possible distributions of g differentiating samples over N/s rounds is a ball-in-bins problem and equals $\binom{N/s - 1 + g}{g}$ (since each round has a sufficient number of 'slots' s to host as many as g balls). The number of ways to choose j rounds in which differentiating samples will be allocated is equal to $\binom{N/s}{j}$. This allocates already 1 differentiating sample in each of the j rounds; j differentiating samples in total. The remaining $g - j$ samples can be freely distributed over the j rounds and this gives another $\binom{j - 1 + (g - j)}{g - j} = \binom{g - 1}{j - 1}$ possibilities (again a balls-in-bins problem). For $1 \leq j \leq g$, we have

$$q_{E=1}(c_1 = j, 0, \ldots, 0) = \binom{N/s}{j} \binom{g - 1}{j - 1} / \binom{N/s - 1 + g}{g}.$$

We will denote this probability by q_j. This allows us to compute trade-off function f in Theorem 4.6 by setting

$$q_E(\sqrt{j}) = q_{E=1}(j, 0, \ldots, 0) = q_j.$$

[‖] We may assume that $S_b = \{(b - 1)ms + 1, \ldots, bms\}$.

We have

$$f(\alpha) = \sum_{j=1}^{g} q_j \cdot \Phi\left(\Lambda(\alpha)\cdot\frac{\sigma}{\sqrt{j}} - \frac{\sqrt{j}}{2\sigma}\right),$$

where function $\Lambda(\alpha)$ is implicitly defined by

$$1-\alpha = \sum_{j=1}^{g} q_j \cdot \Phi\left(\Lambda(\alpha)\cdot\frac{\sigma}{\sqrt{j}} + \frac{\sqrt{j}}{2\sigma}\right).$$

Clearly, $f(\alpha)$ is lower bounded by replacing all $\sqrt{j}$ by the larger $\sqrt{g}$, since this always assumes the worst-case where all g differentiating samples are distributed over different round, hence, composition of $G_{1/\sigma}$ happens g times. This gives

$$f \geq G_{\sqrt{g}/\sigma}$$

and over E epochs this implies the lower bounds $f^{\otimes E} \geq G_{\sqrt{g}/\sigma}^{\otimes E} = G_{\sqrt{gE}/\sigma}$.

Notice that this is exactly the lower bound that follows from Theorem 4.6 for $c^{\mathtt{L}} = \sqrt{g}$ and $l_{c^{\mathtt{L}}} = \sum_{c>c^{\mathtt{L}}} q_E(c) = 0$. Also notice that the lower bound holds for $g > s$. This is because the lower bound assumes the worst-case where all g differentiating samples are distributed over different rounds and this is not restricted by the size of a round (we still have $l_{c^{\mathtt{L}}} = 0$).

If we look at the expression for $q_j = q_{E=1}(c_1 = j, 0, \ldots, 0)$ more carefully, then we observe that

$$\begin{aligned} q_g &= \binom{N/s}{g} / \binom{N/s-1+g}{g} \\ &= \frac{N/s \cdot (N/s-1)\cdot \ldots \cdot (N/s-g+1)}{(N/s+g-1)\cdot(N/s-1+g-1)\cdot\ldots\cdot(N/s)} \\ &\geq (\frac{N/s-g+1}{N/s})^g = (1-(g-1)/(N/s))^g \approx 1-(g-1)g/(N/s), \end{aligned}$$

in other words its probability mass is concentrated in $c_1 = g$. This can be used to extract an upper bound from Theorem 4.5 by setting $c^{\mathtt{U}} = \sqrt{g}$ $(= c^{\mathtt{L}})$ and upper bound the tail $\sum_{c<c^{\mathtt{U}}} q_E(c)$.

For $g \leq s$ we derive

$$\begin{aligned} q_{j+1} &= \binom{N/s}{j+1}\binom{g-1}{j} / \binom{N/s-1+g}{g} \\ &= \binom{N/s}{j}\frac{N/s-j}{j+1}\binom{g-1}{j-1}\frac{g-j}{j} / \binom{N/s-1+g}{g} \\ &= q_j \frac{N/s-j}{j+1}\frac{g-j}{j}. \end{aligned}$$

We have

$$q_j = q_{j+1}\frac{j+1}{N/s-j}\frac{j}{g-j} \leq q_{j+1}\frac{g^2}{N/s-g}.$$

This shows that for $E=1$,

$$\begin{aligned} \sum_{c<c^{\mathtt{U}}} q_E(c) &= \sum_{j=1}^{g-1} q_j \leq \sum_{j=1}^{g-1} q_g(\frac{g^2}{N/s-g})^{g-j} \leq \sum_{j=1}^{g-1}(\frac{g^2}{N/s-g})^{g-j} = \sum_{j=1}^{g-1}(\frac{g^2}{N/s-g})^{j} \\ &\leq \frac{g^2}{N/s-g}/(1-\frac{g^2}{N/s-g}) = \frac{g^2}{N/s-g-g^2}. \end{aligned}$$

We conclude that Theorem 4.5 can be applied for $c^{\mathtt{U}} = \sqrt{g}$ and $u_{c^{\mathtt{U}}} = g^2/(N/s-g-g^2)$ for $g \leq s$ and $E=1$. This yields an upper bound on the trade-off function for $E=1$ that can be composed E times in order to achieve an upper bound for general E.

Lemma G.4. *Batch clipping with shuffling in Algorithm 1 yields a mechanism $\mathcal{M}$ which is h-DP in the strong adversarial model for all pairs of data sets d and d' with with* $\max\{|d \setminus d'|, |d' \setminus d|\} \leq g$ *and* $N = |d \cap d'| + g$ *with*

$$h = G_{\sqrt{gE}/\sigma}.$$

For $g \leq s$, mechanism $\mathcal{M}$ is h-DP for $h = f^{\otimes E}$ with

$$h = f^{\otimes E} \geq G_{\sqrt{gE}/\sigma}$$

for trade-off function f defined by

$$f(\alpha) = \sum_{j=1}^{g} q_j \cdot \Phi(\Lambda(\alpha) \cdot \frac{\sigma}{\sqrt{j}} - \frac{\sqrt{j}}{2\sigma}) \text{ with } 1 - \alpha = \sum_{j=1}^{g} q_j \cdot \Phi(\Lambda(\alpha) \cdot \frac{\sigma}{\sqrt{j}} + \frac{\sqrt{j}}{2\sigma}),$$

where $q_j = \binom{N/s}{j}\binom{g-1}{j-1}/\binom{N/s-1+g}{g}$.

Furthermore, let $c^{\mathtt{U}} = \sqrt{g}$ and $u_{c^{\mathtt{U}}} = g^2/(N/s - g - g^2)$. If $g \leq s$, then for all $\bar{h}$ with the property that $\mathcal{M}$ is $\bar{h}$-DP, we have the upper bound

$$\bar{h} \leq f_{c^{\mathtt{U}}}^{\mathtt{U}\,\otimes E} \leq \hat{f}_{c^{\mathtt{U}}}^{\mathtt{U}\,\otimes E},$$

where

$$\hat{f}_{c^{\mathtt{U}}}^{\mathtt{U}\,\otimes E} = (u_{c^{\mathtt{U}}} + (1 - u_{c^{\mathtt{U}}})G_{c^{\mathtt{U}}/\sigma})^{\otimes E} \approx G_{\sqrt{g}/\sigma}^{\otimes E} = G_{\sqrt{gE}/\sigma}$$

for constant E and small $g^2/(N/s - g - g^2)$. This shows into what extent the lower bound is tight.

Notice that we can again find corresponding Corollaries F.3, F.4, F.5, and F.8.